



Calhoun: The NPS Institutional Archive
DSpace Repository

Theses and Dissertations

1. Thesis and Dissertation Collection, all items

1967-09

Numerical study of secondary effects in flexibility analysis of an ell-shaped piping structure

Miles, John Henry Thomas

Monterey, California. U.S. Naval Postgraduate School

<http://hdl.handle.net/10945/12750>

This publication is a work of the U.S. Government as defined in Title 17, United States Code, Section 101. Copyright protection is not available for this work in the United States.

Downloaded from NPS Archive: Calhoun



<http://www.nps.edu/library>

Calhoun is the Naval Postgraduate School's public access digital repository for research materials and institutional publications created by the NPS community. Calhoun is named for Professor of Mathematics Guy K. Calhoun, NPS's first appointed -- and published -- scholarly author.

Dudley Knox Library / Naval Postgraduate School
411 Dyer Road / 1 University Circle
Monterey, California USA 93943

NPS ARCHIVE
1967
MILES, J.

NUMERICAL STUDY OF SECONDARY EFFECTS IN
FLEXIBILITY ANALYSIS OF AN ELL-SHAPED
PIPING STRUCTURE

JOHN HENRY THOMAS MILES

NUMERICAL STUDY OF SECONDARY EFFECTS IN FLEXIBILITY
ANALYSIS OF AN ELL-SHAPED PIPING STRUCTURE

by

John Henry Thomas Miles
Lieutenant, Civil Engineer Corps, United States Navy
B.S., University of South Carolina, 1961



Submitted in partial fulfillment of the
requirements for the degree of
MASTER OF SCIENCE IN MECHANICAL ENGINEERING
from the
NAVAL POSTGRADUATE SCHOOL
September 1967

LES, J.

1000 M 58576 C.1

ABSTRACT

This thesis considers the influence of four secondary effects on the analysis of a simple ell-shaped piping structure when it is uniformly heated. The four effects are: (a) axial deformation, (b) shearing deformation, (c) beam-column effect, and (d) difference between arc and chord. They are usually neglected in a piping flexibility analysis.

In a paper of several years ago, J. E. Brock developed a theory for this investigation but did not present any numerical results or conclusions. The present thesis reviews Brock's theory and corrects some errors in coefficients. It then describes a digital computer program for IBM System 360 which performs the corresponding calculations. Finally, it draws the new and quite significant conclusion that conventional piping stress analysis, which neglects these secondary effects, can result in gross errors in evaluating stresses in piping configurations which are likely to be encountered in piping practice.

TABLE OF CONTENTS

Section		Page
I	INTRODUCTION	9
	1-1 Background	9
	1-2 Objectives of Work	10
	1-3 Assumptions and Limitations	10
	1-4 Notation	11
II	THEORETICAL ANALYSIS AND INVESTIGATION TECHNIQUE	14
	2-1 Review of Flexibility Analysis for the Ell- Shaped Pipe	14
	2-2 Dimensionless Parameters	19
	2-3 Requirements Imposed on the Digital Computer Program	20
III	DESCRIPTION OF COMPUTER PROGRAM	21
	3-1 General Remarks	21
	3-2 Input	22
	3-3 Loading of Parameter Values	28
	3-4 Combination of Secondary Effects	29
	3-5 Computation by Iteration	30
	3-6 Incrementing the Parameter	32
	3-7 Output	34
	3-8 Subroutines	38
IV	RESULTS OF INVESTIGATION	41
	4-1 Errors in ASME Paper No. 63 - APMW - 21	41
	4-2 Numerical Calculations	42
	4-3 Final Results and Conclusions	42

TABLE OF CONTENTS (cont.)

Section	Page
BIBLIOGRAPHY	52
APPENDIX	
A	COMPUTER PROGRAM NOMENCLATURE
A-1	Main Program Nomenclature 53
A-2	Subroutine ALPHA Nomenclature 57
A-3	Subroutine FACTOR Nomenclature 57
A-4	Subroutine DRAW Description 58
B	COMPUTER PROGRAM FLOW CHART 65
C	DATA CARD DESCRIPTION 67
C-1	Data Card Placement 67
C-2	Sample Data Card Set 69
D	CONTROL CARD ARRANGEMENT 70
E	PROGRAM LISTING 71
E-1	Main Program 71
E-2	Subroutine ALPHA 84
E-3	Subroutine FACTOR 85
E-4	Sample Output 88
F	SAMPLE CALCULATION 89
G	ARC AND CHORD DIFFERENCE 91

LIST OF TABLES

		Page
Table I	Notation	11
Table II	Formulas for the coefficients	13
Table III	Expression relating the quantities of general case with those of the ell-shaped pipe	17
Table IV	Relates Factor to its secondary effect	39
Table V	List of combination designators with associated values of MT & Factor Values	40

LIST OF ILLUSTRATIONS (FIGURES)

		Page
Figure 1	Uniform Tip Loaded Cantilever	14
Figure 2	Ell-shaped pipe experiencing no deformation due to temperature difference	16
Figure 3	Ell-shaped pipe experiencing deformation due to temperature difference	16
Figure 4	Free body diagram of both legs of the ell-shaped pipe under heating	17
Figure 5	Illustration showing the two different parameter sizes	34
Figure 6	Moment/Exact Moment vs. leg ratio for Arc and Chord only suppressed	44
Figure 7	Moment/Exact Moment vs. leg ratio for Arc and Chord only allowed	45
Figure 8	Moment/Exact Moment vs. leg ratio for Shear only suppressed	46
Figure 9	Moment/Exact Moment vs. leg ratio for Shear only allowed	47
Figure 10	Moment/Exact Moment vs. leg ratio for Axial deformation only suppressed	48
Figure 11	Moment/Exact Moment vs. leg ratio for Axial deformation only allowed	49
Figure 12	Moment/Exact Moment vs. leg ratio for Beam column only suppressed	50
Figure 13	Moment/Exact Moment vs. leg ratio for Beam column only allowed	51
Figure G-1	Arc and Chord Difference	91

ACKNOWLEDGEMENTS

A large degree of credit for this thesis is due to Professor John E. Brock of the Mechanical Engineering Department of the Naval Postgraduate School. Some of his work provided the source material for the work reported here and his direction, assistance, and encouragement during the course of this work were contributions invaluable to the author.

The author expresses most sincere appreciation to Professor John E. Brock.

SECTION I
INTRODUCTION

1-1 Background

In 1964 Professor John E. Brock of the Mechanical Engineering Department of the Naval Postgraduate School, Monterey, California, published a paper entitled, "Some Secondary Effects in a Simple Piping Structure Under Heating".¹

The secondary effects referred to are axial deformation, shear, beam column action, and difference between arc and chord. The structural element considered is an ell-shaped piping configuration with a square corner.

In the normal analysis of such a structure, the above mentioned secondary effects are neglected, being regarded as insignificant in relieving the stresses caused by thermal expansion. Only the principal sources of relief, bending and torsion, are usually considered.

Professor Brock undertook to make a complete flexibility analysis of this piping configuration including all of the four secondary effects. The 1964 paper reports his results. It also reports that a computer program had been written to investigate this analysis and the resulting system of equations. The program had been written but was not developed sufficiently to thoroughly investigate the relationships involved.

¹ John E. Brock, "Some Secondary Effects In A Simple Piping Structure Under Heating," Journal of Applied Mechanics, Vol 31, March 1964, pp. 88-90.

1-2 Objectives of Work

The object of this thesis was to write a computer program for a large digital computer that would be capable of performing the calculations required by Professor Brock's theoretical development and one that would allow application of the theory to any engineering problem that involves a piping structure of the given configuration. Further, it attempts to determine under what conditions these secondary effects may become significant in calculation.

1-3 Assumptions and Limitations

The investigations that are reported herein are based on an ell-shaped piping configuration with a square corner. No other shape is considered. Also, the configuration is assumed to have deformation only in the plane of the legs a and b.

It is further assumed that deformations and displacements are small compared to the longitudinal dimensions of the pipe, except that a first approximation is included for the difference between arc length and chord length of the deformed pipes.

1-4 Notation

TABLE I

NOMENCLATURE

a, b	length of straight pipe elements
A	cross sectional area of pipe material
A_F	cross sectional area of pipe contents
e	unit thermal strain
E	Young's modulus of elasticity
F_1, F_2	structural forces in ell-configuration
g_{ij}	coefficients defined in equations (16)-(19)
G	shearing modulus of elasticity
H	$(1-2\nu)\rho^2/E(1-\rho^2)$ = pressure coefficient
I	moment of inertia of pipe cross section
L	length of pipe
M, N	bending moments
p	internal pressure
P	lateral force
Q	axial compressive force
u	$w^{1/2}$
v	$(-w)^{1/2}$
w	QL^2/EI
x	axial coordinate
X	horizontal deflection of joint
y	lateral coordinate
Y	vertical deflection of joint
α_i	coefficients (see Table II)

TABLE I (cont.)

β_1, β_2	coefficients defined in equations (15A) and (15B)
γ	$\zeta P/AG$ = effective shearing strain
δ	tip axial extension
Δ	tip lateral deflection
ϵ_1, ϵ_2	conveniently small positive numbers
ζ	shear distribution factor (see equation (9))
η	$\zeta EI/AGL^2$
θ	rotation of joint
λ	difference between arc and chord (first approximation)
μ_1, μ_2	coefficients in equations (9), (10), and (11)
ν	Poisson's ratio
ρ	(inside diameter of pipe)/(outside diameter of pipe)
φ	tip rotation
ξ	xu/L

TABLE II

FORMULAS FOR α COEFFICIENTS

	$ w < \epsilon_1$	$w > \epsilon_1$	$w < -\epsilon_1$
α_1	$\frac{1}{2} + \frac{5}{24}w + \frac{61}{720}w^2 + \frac{277}{8064}w^3 + \dots$		$(\alpha_4 - 1)/w$
α_2	$\frac{1}{3} + \frac{2}{15}w + \frac{17}{315}w^2 + \frac{62}{2835}w^3 + \dots$	$(\alpha_3 - 1)/w$	
α_3	$1 + w\alpha_2$	$\tan u/u$	$\tanh v/v$
α_4	$1 + w\alpha_1$	$\sec u$	$\operatorname{sech} v$
α_5	$\alpha_1 + \alpha_2 + w\alpha_1\alpha_2 - \frac{5}{12} - \frac{61}{360}w - \frac{277}{4032}w^2 - \frac{50521}{1814400}w^3 - \dots$		$(\alpha_3\alpha_4 - 2\alpha_1)/w$
α_6	$2\alpha_2 + w\alpha_2^2 - \frac{2}{5} - \frac{17}{105}w - \frac{62}{945}w^2 - \frac{17966}{675675}w^3 - \dots$		$(\alpha_3^2 - 3\alpha_2)/w$
α_7	$\alpha_4^2 + \alpha_2 - 2$		
α_8	$\alpha_3^2 - \alpha_2$		
α_9	$\alpha_5 + \eta\alpha_3\alpha_4$		
α_{10}	$\alpha_6 + 2\eta\alpha_8 + \eta^2\alpha_7$		

SECTION II

THEORETICAL ANALYSIS AND TECHNIQUE FOR INVESTIGATION

2-1 Review of Flexibility Analysis for the Ell-Shaped Pipe

Before presenting the investigation of this ell-shaped piping configuration in detail, it is felt that certain principles regarding the analysis merit some review.

If the tip loaded cantilever beam shown in Figure 1 is analyzed by writing a moment equation about point A the following equation results:

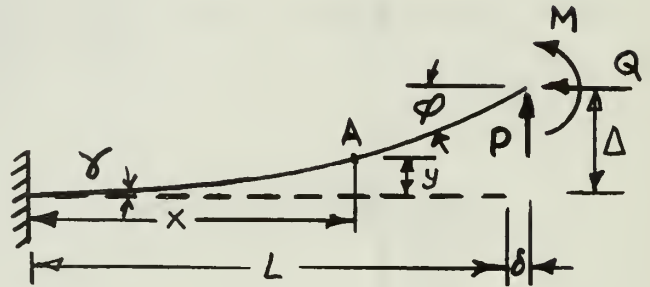


Figure 1

$$(1) \quad EI y'' = M + Q(\Delta - y) + P(L - x)$$

Solving this differential equation and applying boundary conditions yields

$$(2) \quad y = \frac{\{ ML^2 u (1 - \cos \xi) \sec u + PL^3 (1 + \eta u^2) [\sin \xi + (1 - \cos \xi) \tan u] - PL^3 \xi \}}{EI u^3}$$

for boundary conditions

- a. $y(0) = 0$
- b. $y(L) = \Delta$
- c. $y'(0) = \gamma$
- d. $y'(L) = \phi$

See Table I for nomenclature.

In addition

$$(3) \quad \Delta = [M\alpha_1 + PL(\alpha_2 + \eta\alpha_3)](L^2/EI)$$

and

$$(4) \quad \varphi = [M\alpha_3 + PL(\alpha_1 + \eta\alpha_4)](L/EI)$$

The alpha coefficients appearing in the above and subsequent equations are defined in Table II with alternate forms to facilitate computation.

In terms of all four secondary effects, δ is given by equation (5).

$$(5) \quad \delta = eL - \frac{2\nu p L p^2}{E(1-p^2)} - \frac{(Q - A_E p)L}{AE} - \lambda$$

The first term on the right hand side of equation (5) represents thermal expansion, the second represents axial contraction resulting from hoop tension caused by internal pressure, the third represents axial contraction due to axial compression in the metal, and the last represents the effective axial contraction resulting from the difference between arc and chord. The pressure terms may be combined to yield the equation,

$$(6) \quad \delta = (e + p_H)L - QL/AE - \lambda$$

where λ is given by

$$(7) \quad \lambda = \frac{1}{2} \int_0^L ((y')^2 - Y^2) dx$$

when higher order terms are neglected (see Appendix G). If alpha coefficients are substituted,

$$(8) \quad \lambda = (M^2\alpha_8 + 2.0 MPL\alpha_9 + P^2L^2\alpha_{10})(L^3/4E^2I^2)$$

Other pertinent relationships include the shear distribution factor²

$$(9) \quad \zeta = \mu_1 + \mu_2 \rho^2 / (1 + \rho^2)^2$$

where

$$(10) \quad \mu_1 = (7 + 14\nu + 8\nu^2) / 6(1 + \nu)^2$$

and

$$(11) \quad \mu_2 = (10 + 20\nu + 8\nu^2) / 3(1 + \nu)^2$$

To apply the foregoing analysis to an ell-shaped pipe, the corresponding quantities are defined as they appear in that configuration. The basic structure is that shown in Figure 2. If the configuration is subjected to uniform heating, thermal expansion creates a situation resembling that of Figure 3.

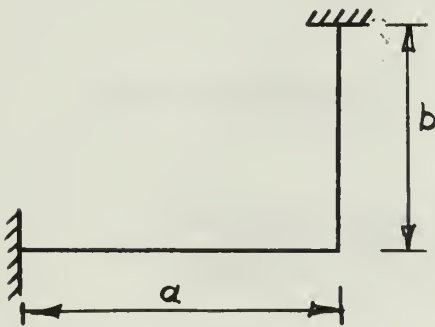


Figure 2

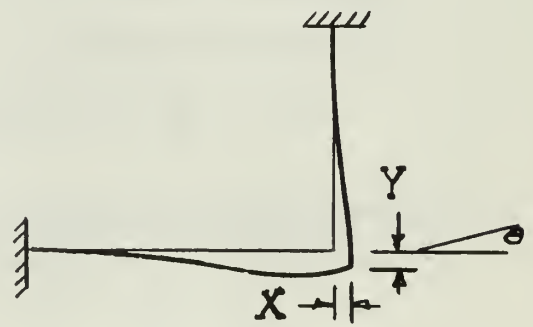


Figure 3

The resulting free body diagrams for the two legs a and b is depicted in Figure 4.

² John E. Brock, "Shear Distribution in Piping," Heating Piping and Air Conditioning, Vol 35, January 1963, pp. 141-143.

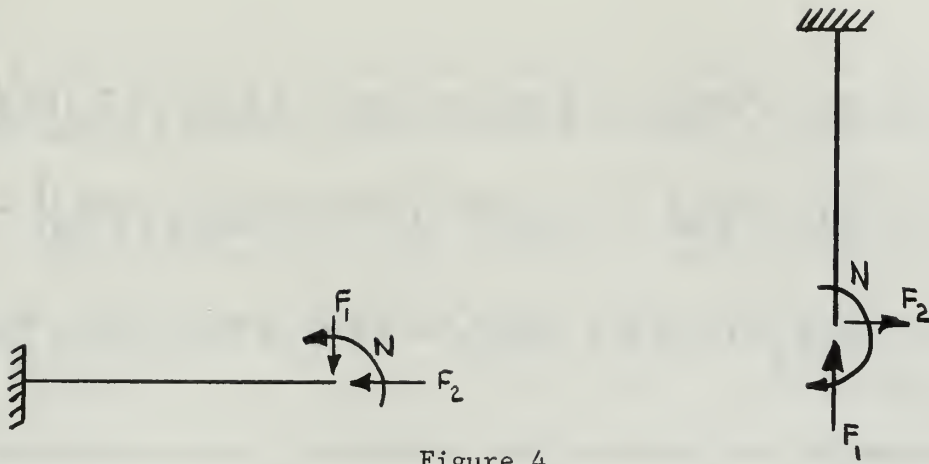


Figure 4

Referring to the horizontal leg as part 1 and to the vertical leg as part 2, Table III list the quantities corresponding to the general case for each part.

TABLE III

<u>ITEM</u>	<u>PART 1</u>	<u>PART 2</u>
L	a	b
P	$-F_1$	F_2
Q	F_2	F_1
M	N	-N
Δ	-Y	X
δ	X	Y
φ	θ	θ

Geometrical continuity requires equating corresponding expressions for X,Y, and θ ; i.e.,

$$\delta_a = \Delta_b$$

$$\delta_b = \Delta_a$$

$$\varphi_a = \varphi_b$$

or

$$(12) \quad ea - F_2 a / AE - \lambda + paH = [-N\bar{\alpha}_1 + F_2 b (\bar{\alpha}_2 + \bar{\eta} \bar{\alpha}_3)] b^2 / EI$$

$$(13) \quad eb - F_1 b / AE - \bar{\lambda} + pbH = [-N\alpha_1 + F_1 a (\alpha_2 + \eta \alpha_3)] a^2 / EI$$

$$(14) \quad Na\alpha_3 - F_1 a^2 (\alpha_1 + \eta \alpha_4) = -Nb\bar{\alpha}_3 + F_2 b^2 (\bar{\alpha}_1 + \bar{\eta} \bar{\alpha}_4)$$

A superior bar indicates that the quantity has been evaluated using the parameters of part 2 and no superior bar indicates evaluations with the parameters of part 1.

Solving equation (14) for the moment, N, yields

$$(15) \quad N = \frac{F_1 a^2 (\alpha_1 + \eta \alpha_4) + F_2 b^2 (\bar{\alpha}_1 + \bar{\eta} \bar{\alpha}_4)}{a\alpha_3 + b\bar{\alpha}_3} = \beta_1 F_1 + \beta_2 F_2$$

where (15A) $\beta_1 = a^2 (\alpha_1 + \eta \alpha_4) / (a\alpha_3 + b\bar{\alpha}_3)$

and (15B) $\beta_2 = b^2 (\bar{\alpha}_1 + \bar{\eta} \bar{\alpha}_4) / (a\alpha_3 + b\bar{\alpha}_3)$

Now define;

$$(16) \quad g_{11} = -\bar{\alpha}_1 \beta_1 b^2 / EI$$

$$(17) \quad g_{12} = [b^3 (\bar{\alpha}_2 + \bar{\eta} \bar{\alpha}_3) + \frac{aI}{A} - \bar{\alpha}_1 \beta_2 b^2] / EI$$

$$(18) \quad g_{21} = [a^3 (\alpha_2 + \eta \alpha_3) + \frac{bI}{A} - \alpha_1 \beta_1 a^2] / EI$$

$$(19) \quad g_{22} = \alpha_1 \beta_2 a^2 / EI$$

and the system of equations become

$$(20) \quad \begin{bmatrix} g_{11} & g_{12} \\ g_{21} & g_{22} \end{bmatrix} \begin{bmatrix} F_1 \\ F_2 \end{bmatrix} = \begin{bmatrix} (e+paH)a - \lambda \\ (e+pbH)b - \bar{\lambda} \end{bmatrix}$$

In order to solve for the unknowns F_1 and F_2 , an iterative procedure is strongly suggested by the nature of the equations.

By assuming initial values of F_1 and F_2 , which can be any value, one can calculate finite values for all the coefficients g_{ij} and the quantities on the right of equation (20). This equation is solved for new values of F_1 and F_2 which in turn provide improved values for the next iteration. This procedure of iteration is continued until satisfactory convergence is reached. Once convergence is obtained such quantities as moment, \emptyset , and other desired values are obtained using the converged values of F_1 and F_2 .

2-2 Dimensionless Parameters

The large number of variables involved in this problem could produce millions of different problems. In order to pursue the investigation as thoroughly as possible and at the same time retain some compactness in the results, four dimensionless parameters were used. These are:

- (1) Leg Ratio a/b
- (2) (Radius of Gyration)/(Total Length)
- (3) Pressure/Modulus of Elasticity
- (4) Thermal Strain

To cover reasonable ranges of all four parameters evidently requires solution of a large number of problems each of which is itself rather demanding. Accordingly, use of a large digital computer is definitely required.

Each of the four secondary effects can either be included or excluded in the calculation. This produces 2^4 or 16 different possible combinations of secondary effects.

2-3 Requirements Imposed on Digital Computer Program

A computer program for studying this problem has certain basic requirements.

- A. First of all, it must be capable of accepting certain input data, which includes a basic physical description of the problem as well as control variables that instruct the program in how to treat these data.
- B. From the input information it must recognize the parameter to be investigated and must load the various values of the parameter over the desired range.
- C. For each parameter value in the given range it must:
 - 1. Solve for the F 's by the iteration procedure and then determine other desired values.
 - 2. Accomplish C.1. for all different combinations of the four secondary effects.
- D. It must increment the parameter value and repeat step C until results for all values in the range have been found.
- E. It must provide results in either tabular or graphical form.

The following section describes the computer program in more detail and shows how the above listed demands are met.

SECTION III

DESCRIPTION OF COMPUTER PROGRAM

3-1 General Remarks

As mentioned in Section II and suggested in reference 2, a computational procedure is evident. Assume initial values of F_1 and F_2 (which may be taken to be zero), evaluate the coefficients g_{ij} and the terms λ and $\bar{\lambda}$ and upon solving the system of equations (20), obtain improved values of F_1 and F_2 , continuing until satisfactory convergence is reached. This idea along with the other basic requirements (A-E of Section II) composes the foundation upon which the computer program is constructed.

The program was first written in Fortran 60 computer language for use on the Control Data Corporation Model 1604 digital computer located at the Naval Postgraduate School, Monterey, California. In the midst of the investigation reported in this paper, the CDC computer was replaced at the Postgraduate School by an International Business Machine Model 360/67. The present version of the program is written in Fortran IV language for use on the latter machine.

All computations in the program are executed using the Double Precision Capability of the system. This feature improves the accuracy of the results but unfortunately requires more time than the single precision calculation.

An effort was made to keep the program as general and as flexible as possible. All of the information needed to specify a given problem is read directly from data cards. The information on the data cards defines the physical condition to be investigated and gives the program additional control information concerning the parameter to be used

in evaluation, range of the parameter, combinations of the secondary effects to be investigated, convergence criteria, etc.

It allows the user four different parameters with which to evaluate the problem. It allows for all sixteen possible combinations of the four secondary effects and computes all sixteen of these for each value of the parameter.

It has optional output. In the data cards the user specifies the desired type of output: tabular, graphic, or both.

To describe the computer program more completely requires explaining in more detail the functions of its seven major parts.

- A. Input
- B. Loading of Parameter Values
- C. Consideration of All 16 Different Combinations
- D. Computation by Iteration
- E. Incrementing Parameter to Next Value in Range
- F. Output
- G. Subroutines

3-2 Input

The term input is used here in referring to the information that is supplied to the program via the data cards. In all, there are fifty-five different items read in as data. These items completely define the physical situation under consideration. Further, it informs the program of the dimensionless parameter to use in the investigation, the range of the parameter, conditions to investigate, and so on. Each of the variables read in as data is listed below along with a description of its function in the program. The order of this listing is the order in which they appear on the data cards. The names of the variables are the names which are used as input. The same physical

quantity may be called by other names later in the program and it should be remembered that the variable name mentioned below is applicable only in the input section of the program.

R	length of the shorter leg, a, (inches)
S	length of the longer leg, b, (inches)
ZIN	pipe inside diameter (inches)
ZOUT	pipe outside diameter (inches)
TEMP	temperature difference (degrees fahrenheit) = $(T_{\text{final}} - T_{\text{initial}})$
Q	internal pressure (psi)
E	modulus of elasticity of pipe material (psi)
COEFI	coefficient of linear expansion of pipe material (in/in - °F)
KPAR1	Integer variable which can assume the integer values 1 to 4 inclusive, that designates the parameter to be used in the investigation as follows: KPAR1 = 1 = Leg Ratio KPAR1 = 2 = (Radius of Gyration)/(Total Length) KPAR1 = 3 = Pressure/Modulus of Elasticity KPAR1 = 4 = Unit Thermal Strain
L	Integer variable that specifies the size of the major parameter increment (see Section 3-3). This can be any value but for graphic output should not require more than thirty stored values of the parameter.

- NN Integer variable that designates the initial value of N, itself an integer variable which is used as a subscript for the parameter values. This (the value of NN) is useful if the program ever exceeds the time allotted for one run and it is terminated by the operator, or if the given size of increments do not yield convergence over the desired range. By giving NN the value of the last value of N on the terminated run, the user is able to modify certain other control variables such as L, LL1, FTE's, etc., for the next computer run and begin computation at the point where the last run ended, regardless of the reason for termination. For the first run of a given parameter or problem the value of NN is always 1.
- MA Integer variable that designates the final value of the subscript N in the parameter range.
- MB Integer variable which designates the initial value of MT. The significance of MT is explained in more detail in the section defining loading. MB can take on any value from 1 to 16.
- MC Integer variable that indicates the final value of MT. $MC \geq MB$ always and $1 \leq MC \leq 16$. Both of the above conditions must be satisfied. The program is designed in such a way that if MT does not equal its final value, MC, then MT will be incremented by 1, or the next combination of secondary effects will be calculated using the same parameter value. What

this in effect does is have the computer make all the calculations for all the combinations between and including $MT = MB$ and $MT = MC$. For example, if $MB = 3$ and $MC = 7$ for a given set of input values, MT would take on values 3, 4, 5, 6, and 7 for each value of the parameter. Consequently, the output would reveal results for the combination corresponding to the 5 different values of MT .

MD Integer variable that designates the type of output desired.

1 ≤ MD ≤ 3

MD = 1 Tabular output only

MD = 2 Graphic output only

MD = 3 Both tabular and graphic output

MIT Integer variable that indicates the limit to the number of iterations the program will perform. If this number is reached before the convergence criterion is met, the program will terminate operation on its current set of data, give the necessary output, then proceed to the next set of data.

KKK Integer variable that assigns a value to the constant IA . IA is used in a relationship that determines the improved values of F 's to be used in the next iteration on the basis of the last assumed values of the F 's. The relationship involved is:

$$F_1 = \frac{4 \times F_1(\text{NEW}) + IA \times F_1(\text{OLD})}{4 + IA}$$

and a similar relationship exists for the improved F2. It was found that generally as the value of N increases IA should increase also, thereby making the improved value closer to the old value than the new value.

LL1 Integer variable that specifies the size of the sub-increments of the parameter. (See section concerning loading).

EP2 This value sets the convergence criteria. After each iteration the new values of the F's are compared to the old ones using the equation

$$\frac{|F1(NEW) - F1(OLD)| + |F2(NEW) - F2(OLD)|}{\sqrt{(F1(NEW))^2 + (F2(NEW))^2}} \leq EP2$$

If this is satisfied the values are considered converged.

EXSCAL This quantity is the argument used in the DRAW subroutine (see Appendix A-4) that assigns the scale to the X-axis or abscissa axis when graphic output is desired.

YSCALE Same as EXSCAL above except it governs scale on Y-axis.

IXUP Another argument for the DRAW subroutine that specifies how many inches up the graph one wants the X-axis located.

IYRIGH Specifies how many inches from the left side of the graph one wants the Y-axis.

FTE11-FTE162 The 32 different values are the last converged values of F1 and F2 for a particular combination of secondary effects. The final digit distinguishes between F1 and F2. The other numerical digits run from 1 through 16 and correspond to values of MT. These FTE values

are retained by the computer and used as the first approximation for the F 's in the corresponding combination when the parameter has been incremented. In most investigations the values assigned by the data card are all zero. These values, however, are just the initial values assigned to the FTE's. Once computation begins the computer takes over the task of assigning these values. Every time convergence is reached the values of two FTE's change and as convergence is reached for all combinations and one parameter value, all 32 change their values. Use of the variables aids the computation process in two ways. The nature of the governing equations causes difficulty with convergence under some conditions. Therefore, if the first assumed values of the F 's are near the correct values a better chance for convergence exists. If, on the other hand, the guess is arbitrary, or not close to the converged value, the system of equations may not converge. If the last converged values are assumed as the first approximations with the incremented parameter, they should be good estimates depending on the size of the parameter increment. The smaller the parameter increment the better the estimate should be.

Another utilization of the 32 variables allows the user the capability of picking up computation at any point in a parameter range. Suppose, due to excess time the run was terminated before the entire range was completed. Then, on the subsequent run, the user

would be able to start at the same place in the parameter range that the previous run ended. This provides continuity without duplication and the ability to read reasonable first approximations to the F's in as data regardless of where in the range one desires to continue.

3-3 Loading of Parameter Values

The integer variable KPAR1, read from the data card, instructs the program which dimensionless parameter will be used to investigate the structure. For a given set of data (one set of six data cards) only one parameter can be investigated. Each of the four values that KPAR1 may take on indicates a different loading scheme for the six subscripted variables. Once the program knows the value of KPAR1 it proceeds to the proper loading scheme for that parameter.

The six subscripted variables involved are:

AA(N)	leg a
BB(N)	leg b
XDIN(N)	pipe inside diameter
XDOUT(N)	pipe outside diameter
XP(N)	internal pressure
ALIT(N)	unit thermal strain

Here again the variable names associated with the physical quantities apply only in the loading scheme. The same physical quantity may have another variable name at another place in the program. See Appendix A for a complete listing of computer program nomenclature.

The subscript N takes on integer values from 1 to MA. For each value of N, the six subscripted variables have values stored in the computer memory. In each of the different loading schemes only one

of the six subscripted variables takes on different values for each different value of N. The other five have fixed values.

Associated with N in the loading procedure are two other integer variables, MA and L. The values of these two quantities determine the range of the parameter and the size of the main parameter increment.

An important feature of the loading process is that the parameter values are loaded in such a way as to produce the least critical loading conditions on the structure for the parameter value associated with $N = 1$ to the most critical loading condition when $N = MA$. This essentially, means that at the least critical end of the range the initial assumption of the arbitrary values of the F's will probably yield convergence. However, as the parameter proceeds to the more critical end of the range, the facility utilizing the FTE's becomes more and more necessary in achieving convergence.

3-4 Combination of Secondary Effects

The capability that allows the computer program to solve the system of equations for all sixteen different combinations of secondary effects for each parameter value is afforded through the subroutine FACTOR. To supply the subroutine with information it needs, the integer variable MT is used. MT takes on values 1 - 16 for each parameter value in the range. Each of the 16 values represents a different combination of secondary effects. When the subroutine is called with MT specified, it responds by supplying the main program with four factors and a particular designator. These four factors are multiplied by the terms in the system of equations that represent the four different secondary effects. Each of the factors can have only two different values: 1.0 and 0.0. It is therefore evident that if the combination requires the suppression of an effect in the calculation, the governing factor is

zero; on the other hand if the effect is to be included the governing factor is unity.

The main program is so designed that all of the sixteen different combinations do not have to be calculated each time, say in a situation where the user is only interested in one or two combinations. If such is the case the integer variables MB and MC are manipulated to provide the combinations desired. MB sets the initial value of MT. After each combination is completely solved for all desired quantities the main program tests the value of MT against that of MC. MC represents the final value of MT to be considered for each parameter value and in the test if MT is less than MC its value is incremented by 1 and that combination is investigated. This process continues until $MT = MC$ and after the $MT = MC$ combination is solved the program increments the parameter. The limitation on this capability lies in the fact that combinations represented by values of MT between $MT = MB$ and $MT = MC$ will be calculated even if some of these intermediate cases are not specifically wanted.

The designator referred to earlier is realized as a label in the printout of the results. It is printed in a certain column of the results and is a four character identifier that tells the value of each factor in that particular combination (see the explanation of this subroutine for specific meanings for these designators in Section 3.8).

3-5 Computation by Iteration

The iterative procedure, as mentioned earlier, is that of assuming initial values of F_1 and F_2 (which may be taken to be zero), evaluating the coefficients g_{ij} and the terms λ and $\bar{\lambda}$ and upon solving the system of equations (20), obtaining improved values for the F 's. This is continued until satisfactory convergence is reached. Through the use

of the values of the FTE's an attempt is made to minimize the number of iterations required. This saves computer time as well as increases the probability of convergence.

After each iteration the newly calculated values of the F's are tested against the old ones using the following convergence criterion:

$$\frac{|F_1(\text{NEW}) - F_1(\text{OLD})| + |F_2(\text{NEW}) - F_2(\text{OLD})|}{\sqrt{(F_1(\text{NEW}))^2 + (F_2(\text{NEW}))^2}} \leq \text{EP2}$$

If this criterion is not satisfied, the program selects improved values for the next iteration and sends these newly selected assumptions of F1 and F2 through the system of equations again. The process for the selection of the new F's is defined by the following two relationships:

$$F1_{\text{NEXT}} = \frac{4 \times F1_{\text{N}} + \text{IA} \times F1}{4 + \text{IA}} ; \quad F2_{\text{NEXT}} = \frac{4 \times F2_{\text{N}} + \text{IA} \times F2}{4 + \text{IA}}$$

where F1NEXT and F2NEXT are the improved values of F1 and F2 respectively to be used in the next iteration. F1N and F2N are the new values of F_1 and F_2 just computed and F1 and F2 in the equations are the values of F_1 and F_2 assumed at the beginning of the iteration. The factor IA is an integer whose value is set by the integer variable, KKK, read in from the data card. It is obvious that the improved value can be closer to the older value or the newer value at the discretion of the user. The nature of the equations is such that when the parameter value produces less critical loading conditions, convergence is achieved more rapidly if the improved value is taken to be closer the new values. Conversely, when the parameter is in a more critical region, a selection of an improved value closer to the old value produces convergence

more rapidly. The term "critical region" refers to that range of the parameter where the system of equations behave in a highly nonlinear fashion.

There is no set rule for this selection of improved values and the best instruction is gained through experience with the program. Using values of IA between 0 and 10 is normally adequate for most situations.

If the convergence criterion is satisfied, the latest calculated values of the F's are accepted as the correct ones for the given value of the parameter and the given combination of secondary effects. Using these values of the F's other necessary quantities are calculated. The program then goes to the next combination of secondary effects and begins iteration for the solution of the F's for that situation. The first approximations for the F's are the converged values found for that combination with the previous parameter value. When the program has solved the equations for all combinations desired, it increments the parameter and the process starts all over again.

3-6 Incrementing the Parameter

Two different parameter increments are utilized in the program. They are referred to hereafter as Δ_{PAR} and δ_{par} . The two can be described as follows:

If $\text{PAR}(N)$ is the parameter for a particular value of the subscript, then $\text{PAR}(N + 1) = \text{PAR}(N) + \Delta_{\text{PAR}}$. Therefore, Δ_{PAR} is the increment whose size is the difference between two consecutive stored parameter values. The size of Δ_{PAR} is determined by appropriate input data and by the control variable L (see description of L in Section 3-2).

The sub-increment, δ_{par} is used to give the program the capability of making the parameter as small as deemed necessary in order to achieve convergence. The basic theme of the investigation of any of the four dimensionless parameters is to proceed from that end of the range that provides the least critical load toward the end which produces the most critical condition in the range. Each time the parameter is incremented, the last converged values of the F's are used as the first approximation. By making the increments smaller, the first approximation is closer than if the parameter had a larger increment. Therefore, it is conceivable that by making these increments small enough, convergence can always be realized. It is for this reason that the program uses increments of two different sizes.

For the graphic output capability, it is necessary to store many values of the variables that are used in the graphs. The storage capacity of the computer thus establishes a lower limit for the larger increment. In order to further decrease the size of the increment the smaller increment, δ_{par} , is used. Except when they coincide with Δ_{PAR} values, parameter values resulting from this smaller increment and the corresponding calculated values are not stored, but used only to obtain new FTE's for the subsequent parameter value. Another way of expressing the process is to say that after all necessary calculations are made for one parameter, the parameter value is incremented by δ_{par} . Using the last converged values of the F's as first approximations, converged values for the new parameter value are sought. Thus the incrementation continues over the parameter range. If the first parameter value considered was a stored value, then after LL1 sub-increments the next stored parameter value will be reached. LL1 is the integer variable read from the data card.

Figure 5 aids in understanding the use of the two increments.

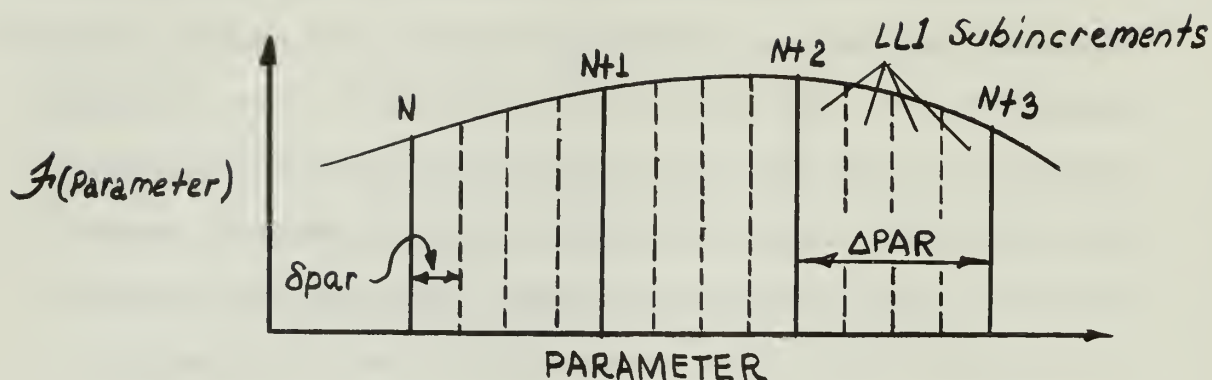


Figure 5

From Figure 5 the following relationships are apparent:

$$\text{LL1} \times \delta \text{par} = \Delta \text{PAR} \quad \text{PAR}(N + 1) = \text{PAR}(N) + \Delta \text{PAR}$$

Also, for the parameter values corresponding to the solid vertical lines, necessary calculated values are stored for graphic or tabular output. For parameter values corresponding to the broken vertical lines only converged values of the F's are calculated for use as the first approximation when the parameter is incremented.

To summarize, each time the parameter is incremented it is increased by the amount δpar . Every LL1 times, all calculated values necessary for output are stored. The size of ΔPAR is somewhat restricted by the storage capacity of the computer but the size of δpar has no restrictions. It must be remembered, however, that by decreasing the size of δpar , the computing time for a given range of the parameter is increased accordingly. Therefore, in the interest of time on the computer, the value of δpar should be as large as possible, commensurate with providing the necessary convergence.

3-7 Output

The user has some choice concerning the type of output. The results can be represented in tabular form, graphical form, or both.

To specify the type of output desired for a given set of data, assign the proper value to the input integer variable MD. MD can take on values 1 to 3 with the following significance attached to each:

MD = 1	Tabular Output only
MD = 2	Graphic Output only
MD = 3	Both Graphic and Tabular Output.

If MD is set equal to 1, ten columns of calculated values will be printed out for all of the desired combinations of secondary effects. The values shown are: the value of the subscript (N), the value of the integer MT, the combination designator, moment, moment ratio, the converged values of F_1 and F_2 , the value of the parameter, and the number of iterations required for convergence in each case. In addition at the top of each page of output are listed the values of the raw data exactly as it is read in from the data card. These values only serve as a convenience in showing what was read into the program and what the individual values were prior to the parameter loading. Other values that are printed include the exact moment, the parameter that was investigated, the elapsed time in seconds since the last stored parameter value with its associated values were calculated, and the critical loads for each parameter except the leg ratio. Actually the output sheet is very straightforward if one follows the column labels.

Some of the above mentioned terms deserve a little more explanation.

Combination Designator: The four character designator supplied to the main program by the subroutine Factor. The four characters are either zero or one and they represent the value of a particular factor for the given combination of secondary effects. For instance, if the

designator is 1001, the factor that is multiplied by the term representing the secondary effect, arc and chord difference, is 1. The factor that is multiplied by the secondary effect term for shear is 0. The factor multiplied by the term representing axial deformation is 0 and the factor multiplying the beam column term is 1. The order of the factors always appear in the same order. Arc and chord, shear, axial deformation, beam column action appear in that order from left to right.

Exact Moment: When the term exact is used referring to a calculated quantity in the system of equations it means that the calculation was made with all four of the secondary effects allowed and none of the effects suppressed; for such a calculation, the combination designator is 1111.

Moment Ratio: Ratio of moment calculated with given combination designator to exact moment.

If MD is set equal to 2, only graphic output results. For graphic output certain limitations exist:

- a. The ordinate is always the moment ratio.
- b. The abscissa is always the parameter under investigation.
- c. MA or the number of stored values for each subscripted variable should not exceed 30.

One graph is obtained for each set of data investigated. On each graph six sets of values are plotted. These six sets correspond to the first six values of MT. See Subroutine Factor in Section 3.8 for the combinations of secondary effects that correspond to the six values. The combinations included in the six are with all effects allowed (exact solution), all effects suppressed, four combination where only 1 of the four effects is allowed and the other three are suppressed. Therefore, if the user specifies graphic output only, the input integer

variables MB and MC should equal to 1 and 6 respectively. Calculations for all other combinations would be redundant since there is no tabular output and the first six combinations are all that will be represented on the graph. If on the other hand, tabular output is selected or the user desires both types of output on the same run then the above restriction is not applicable.

Before one employs the graph plot form of output it is necessary to have some knowledge of the values that are to be plotted. The data card contains four of the arguments to the DRAW subroutine. For a meaningful graphical output a reasonable scale has to be selected and other parameters have to be chosen. The recommendation therefore is to use the tabular output first and from it plan the graphical output so as to describe the results most usefully. In addition the user should be familiar with the information relating to the DRAW subroutine contained in the next section and in the subroutine system library write-up reproduced in Appendix A-4.

3-8 Subroutines

There are three subroutines used in the program. The first two subroutines, designated ALPHA and FACTOR were programmed by the author and the third, designated DRAW, is from the subroutine library at the computer facility of the Naval Postgraduate School, Monterey, California. The complete library write-up for DRAW and its use appears in Appendix A-4.

ALPHA: This subroutine is used by the main program to compute the values of the alpha coefficients as they are shown in Table II, page 13. It is called twice during each iteration and supplies the program with values of the α_i or α_s corresponding to the information that is given to it, namely, ω, η , and EP1 or $\bar{\omega}, \bar{\eta}$, and EP1,

depending on whether the main program wants the values of α 's or $\bar{\alpha}$'s.

FACTOR: This subroutine is used by the main program to set the combination of secondary effects to be considered. As previously mentioned, it works along with the integer variable MT which takes on values 1 through 16. Each of the values designates a certain combination of the secondary effects. Upon receiving this number designation, the subroutine supplies the program with four factors each of which has the value of either 1.0 or 0.0 depending on the combination. These factors are multiplied by the quantities that introduce the secondary effects into the system of equations. If the factor is 1.0 that particular secondary effect is included in the calculations; if it is 0.0 the secondary effect is suppressed in the calculation. Tables IV and V on the following two pages give more information concerning these combinations and their designations that is helpful in understanding the particular system for varying the combinations.

DRAW: This subroutine is the one through which the program affords the user the graphic output option. The complete write-up of this subroutine just as it appears in the Postgraduate School's subroutine library is reproduced in Appendix A-4. It is felt that no further explanation is necessary except for the four arguments that are read in as data in the main program. Referring to the write-up in Appendix A-4, the arguments h. through k. are supplied to the calling statements by the last four values on the second data card. This allows user control of the scales desired and the axis location.

TABLE IV

FACTOR NAME	SECONDARY EFFECT THAT IT CONTROLS
FAC1	Difference in Arc and Chord
FAC2	Shear
FAC3	Axial Deformation
FAC4	Beam Column Action

TABLE V

<u>MT</u>	<u>DESIG- NATOR</u>	<u>STATUS OF EACH SECONDARY EFFECT</u>				<u>VALUE OF EACH FACTOR</u>			
		<u>A-C</u>	<u>SH</u>	<u>AX-DEF.</u>	<u>B-C</u>	<u>FAC1</u>	<u>FAC2</u>	<u>FAC3</u>	<u>FAC4</u>
1	1111	A	A	A	A	1.0	1.0	1.0	1.0
2	1000	A	S	S	S	1.0	0.0	0.0	0.0
3	0100	S	A	S	S	0.0	1.0	0.0	0.0
4	0010	S	S	A	S	0.0	0.0	1.0	0.0
5	0001	S	S	S	A	0.0	0.0	0.0	1.0
6	0000	S	S	S	S	0.0	0.0	0.0	0.0
7	1010	A	S	A	S	1.0	0.0	1.0	0.0
8	0110	S	A	A	S	0.0	1.0	1.0	0.0
9	0011	S	S	A	A	0.0	0.0	1.0	1.0
10	0101	S	A	S	A	0.0	1.0	0.0	1.0
11	1001	A	S	S	A	1.0	0.0	0.0	1.0
12	0111	S	A	A	A	0.0	1.0	1.0	1.0
13	1011	A	S	A	A	1.0	0.0	1.0	1.0
14	1101	A	A	S	A	1.0	1.0	0.0	1.0
15	1110	A	A	A	S	1.0	1.0	1.0	0.0
16	1100	A	A	S	S	1.0	1.0	0.0	0.0

A means effect is allowed

B means effect is suppressed

SECTION IV

RESULTS OF INVESTIGATION

4-1 Errors in Paper No. 63--APMW-21

Prior to developing the computer program to utilize the relationships of Section II, certain review and checking was undertaken by the author. This review was undertaken principally to gain a working familiarity with the equations and nomenclature. During the course of this review and later in developing the program a few errors were detected in the original paper.

Most of these mistakes were algebraic in nature. One was in the area of the critical buckling loads for beam column considerations. The mistakes are:

1. Equation (8), the factor "2.0" in the second term of the numerator did not appear in the same equation in Brock's paper.

There were three mistakes in the formulas for the α coefficients of Table II.

2. & 3. In the formula for α_6 , the α 's in the first two terms of the series should be α_2 's. In the reference paper they are α_1 's.

4. The formula for α_{10} , the last term should be $\eta^2 \alpha_7$ as opposed to " $\alpha^2 \alpha_7$ " in the paper. This was apparently a typographical error.

5. The last incorrect item found was in the statement relating to the critical buckling loads. The paper indicates that $w = \pi^2/4$ defines the critical buckling load for the ell-shaped pipe structure. This is true only for the equal leg case but is not true in general.

All of the results reported in this paper reflect the corrections as stated above.

4-2 Numerical Calculations

With the use of the digital computer program described in Section III, calculations were made for several sizes of pipe and for various values of temperature increase, as well as for different values of the other inputs. This was done not only to test the integrity of the computer program, but also to obtain sufficient data to permit a search for trends and useful modes of presentation. It turns out that the leg ratio, a/b , is the most significant parameter, as might have been expected, and presentations are confined to displaying the moment ratio as a function of leg ratio for various combinations of secondary effects all curves shown being for the following specific case.

Length of leg a	varied
Length of leg b	360 inches
pipe inside diameter	10.020 inches
pipe outside diameter	10.750 inches
Temperature difference	530 degrees fahrenheit (70-600 degrees F)
pipe internal pressure	400 psi
pipe modulus of Elasticity	27.4×10^6 psi
pipe coefficient of linear expansion	7.23×10^{-6} in/in deg F

4-3 Final Results and Conclusions

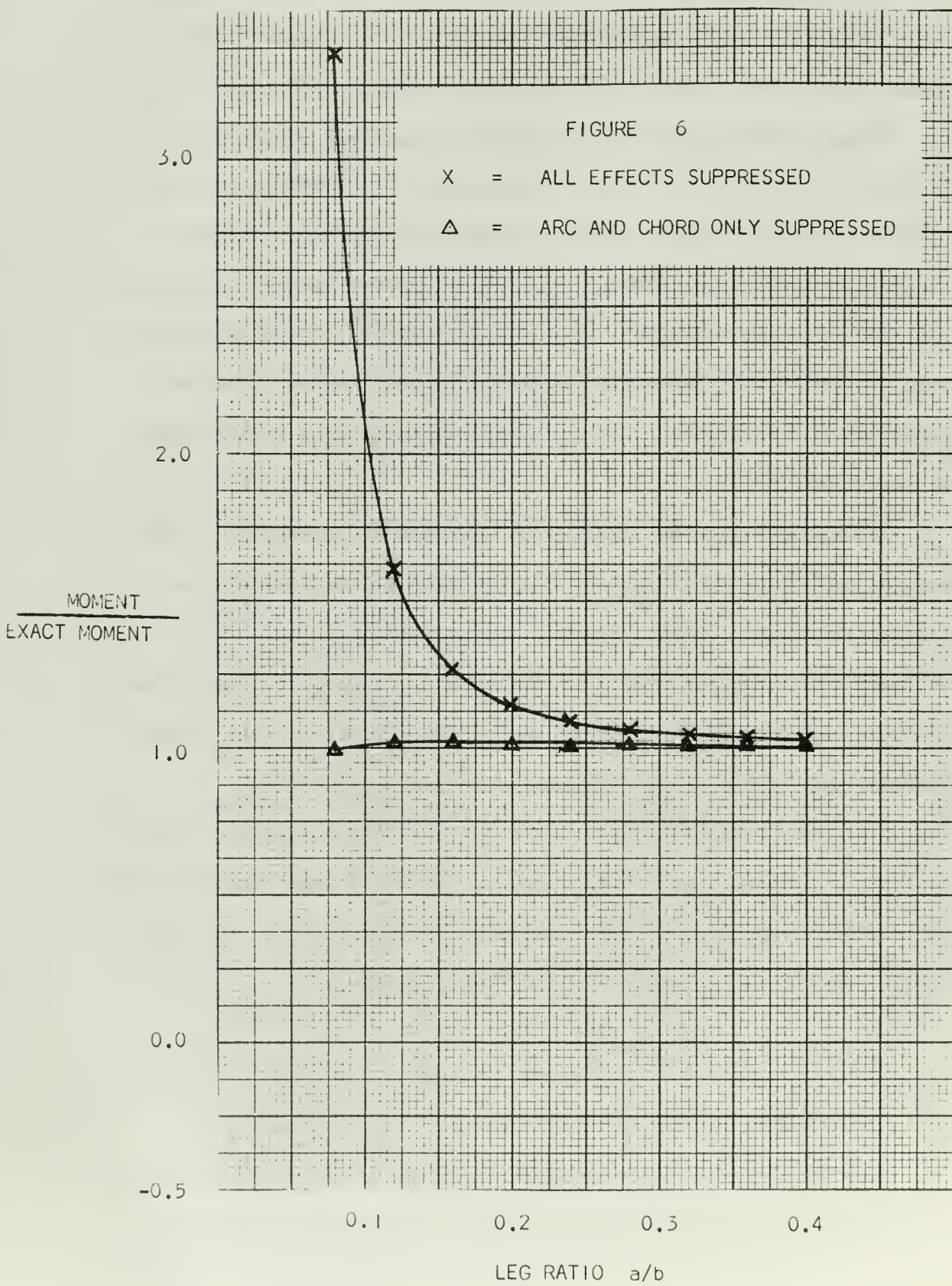
The results are shown in the form of eight curve sheets, Figures 6 through 13, which should be considered in pairs, the first of each pair showing the effect of suppressing (i.e. failing to consider) only a single secondary effect and the second showing the

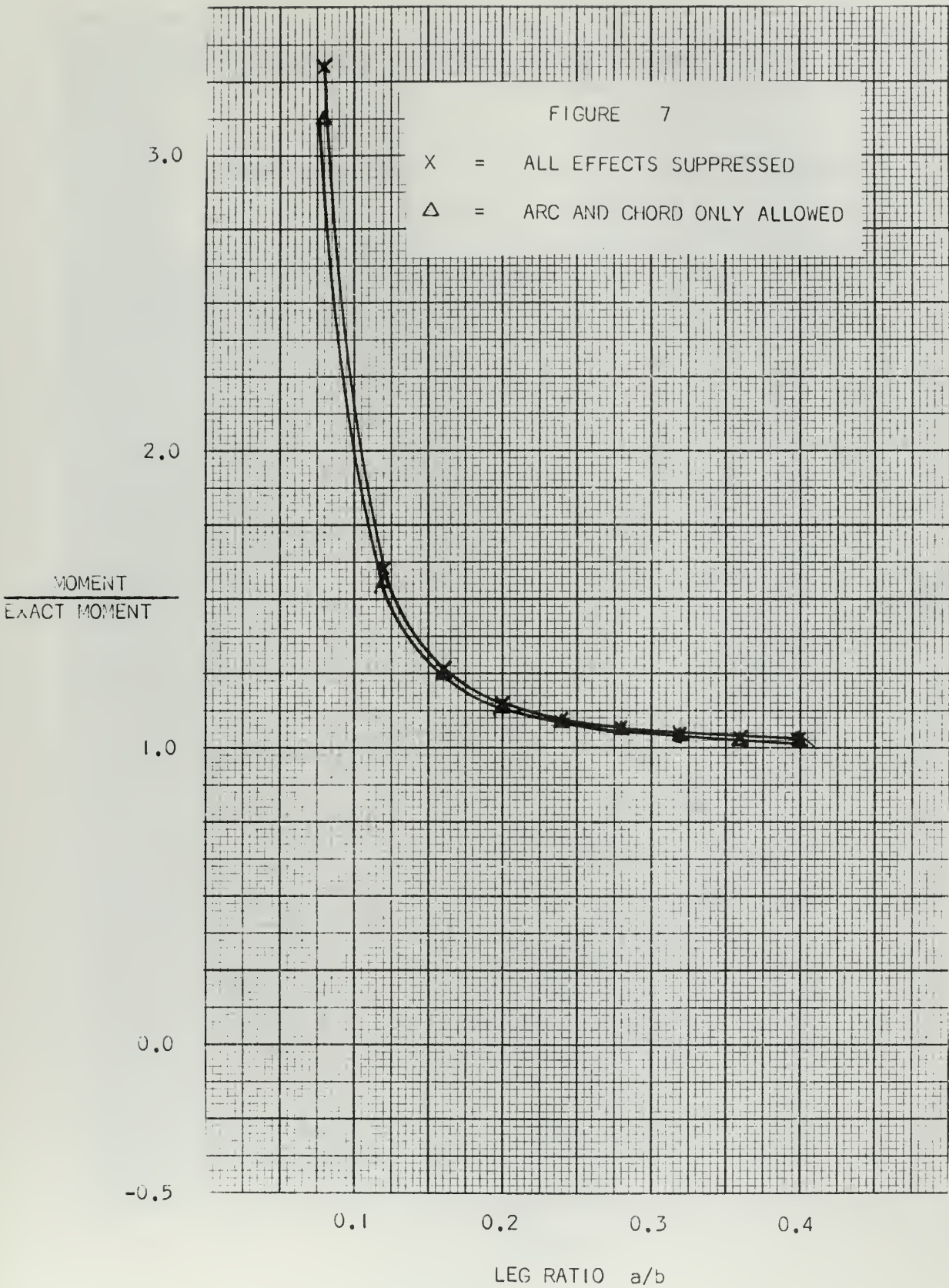
effect of suppressing the other secondary effects and considering only this one. For comparison purposes, each curve sheet also shows the effect of neglecting all four secondary effects.

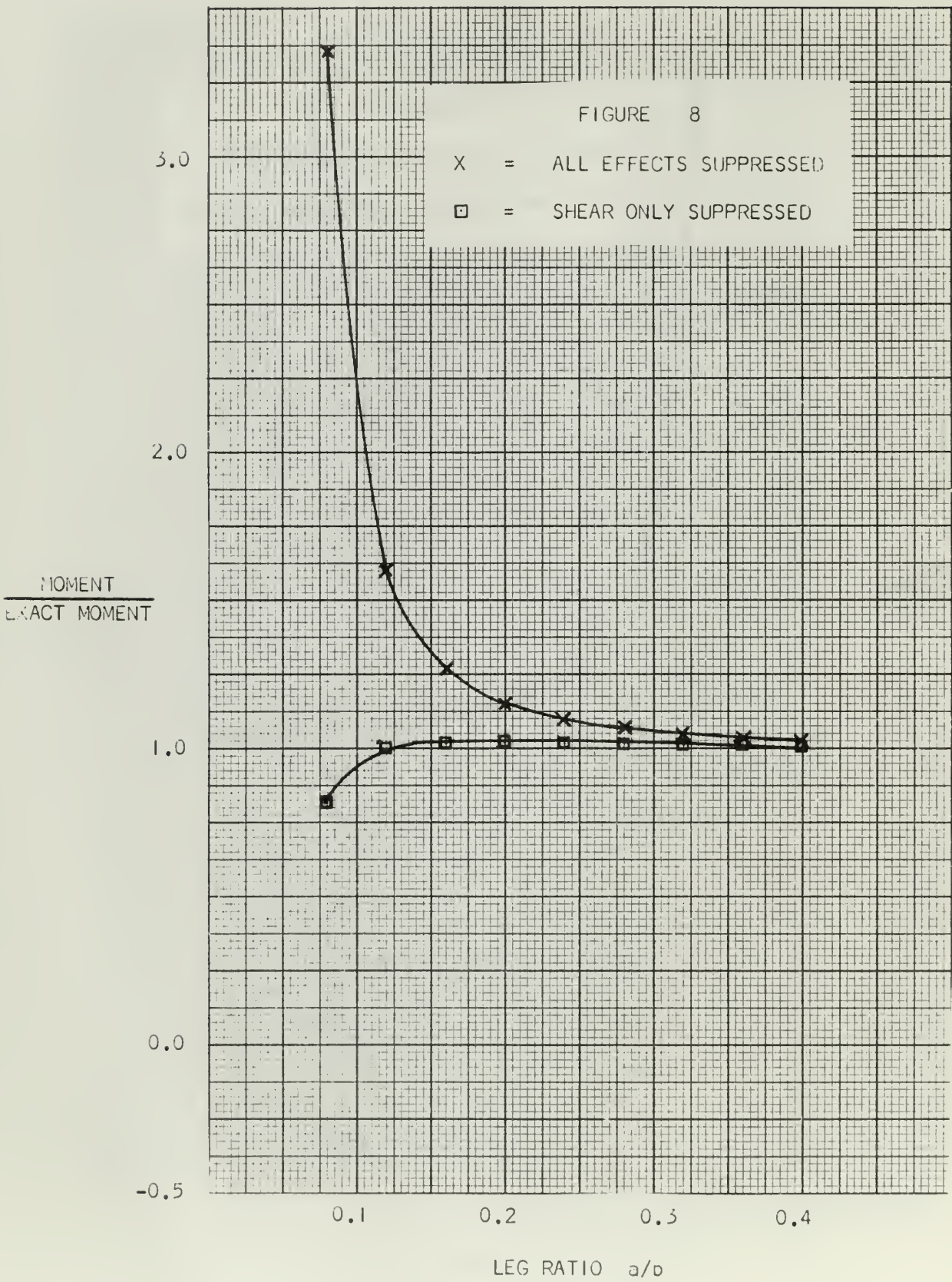
These curves are based only upon what has been called the "moment ratio", i.e., the ratio of the moment at the intersection of the two legs as calculated by an inferior procedure to that calculated by the best available procedure, namely that of including all four secondary effects. It is clear that substantially the same sort of result would have been obtained by considering a similar moment ratio for either of the two built-in ends or a similar ratio of reaction force.

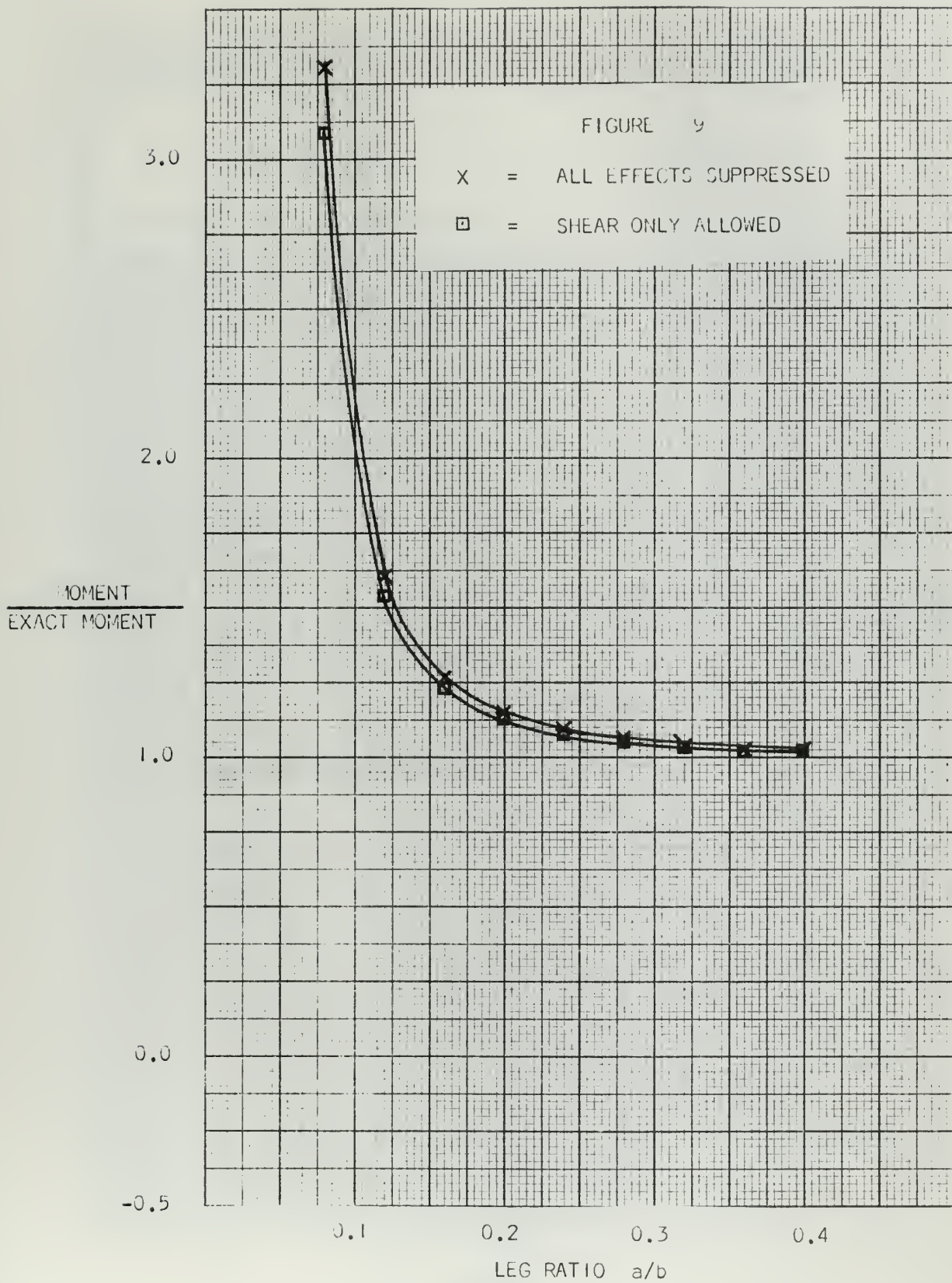
These curves may be regarded as speaking for themselves. The most significant conclusion that may be made is that neglecting secondary effects, indeed can have a profound influence on the accuracy of piping flexibility calculations, leading to gross errors in the evaluation of stresses in cases of leg ratios which, though extreme, are likely to be encountered in piping practice.

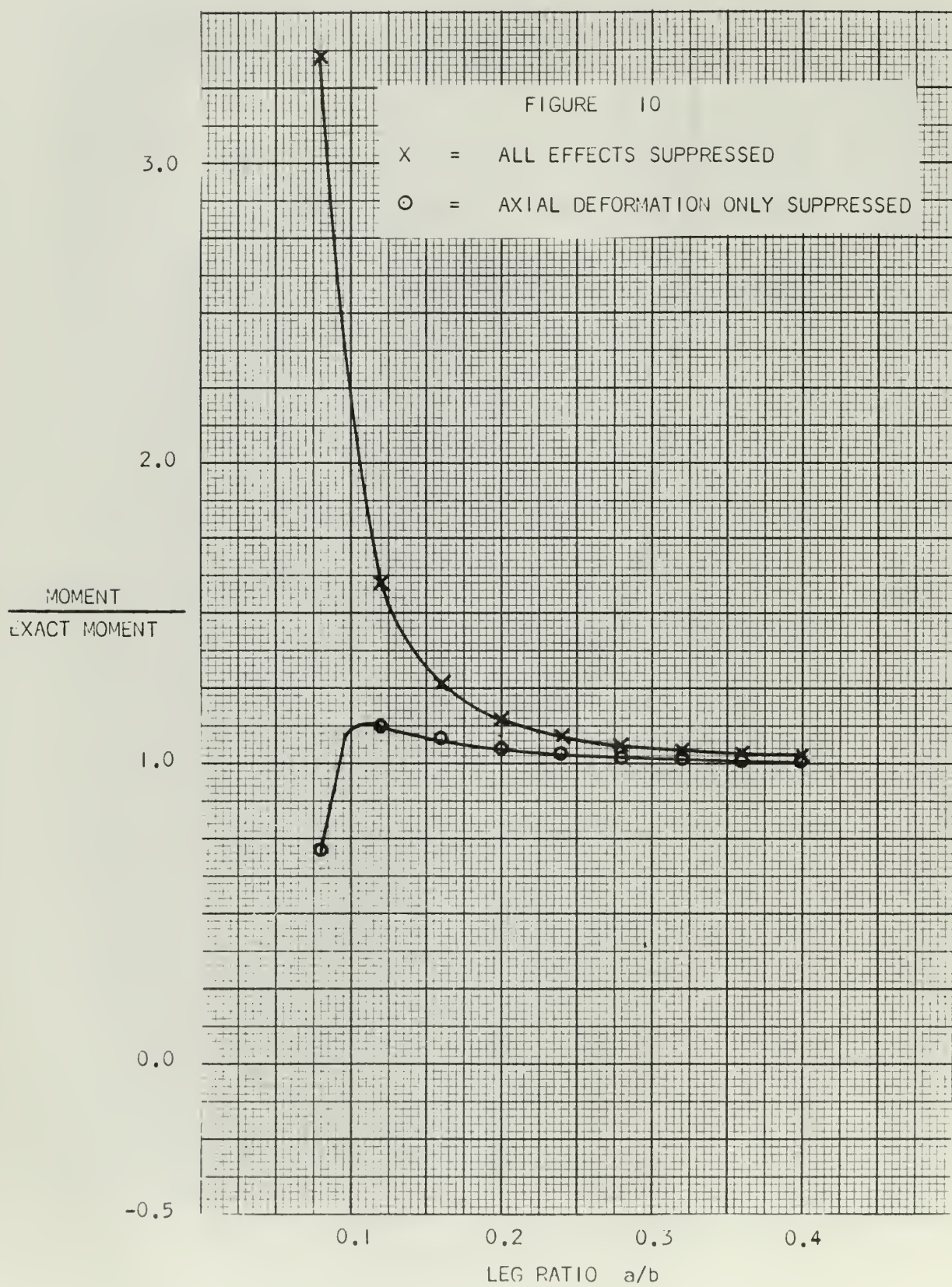
Since the secondary effects are quite subtle to evaluate, this conclusion could not have been forthcoming from a less sophisticated and demanding approach. It is believed to be an entirely new conclusion and it is certainly a significant one.

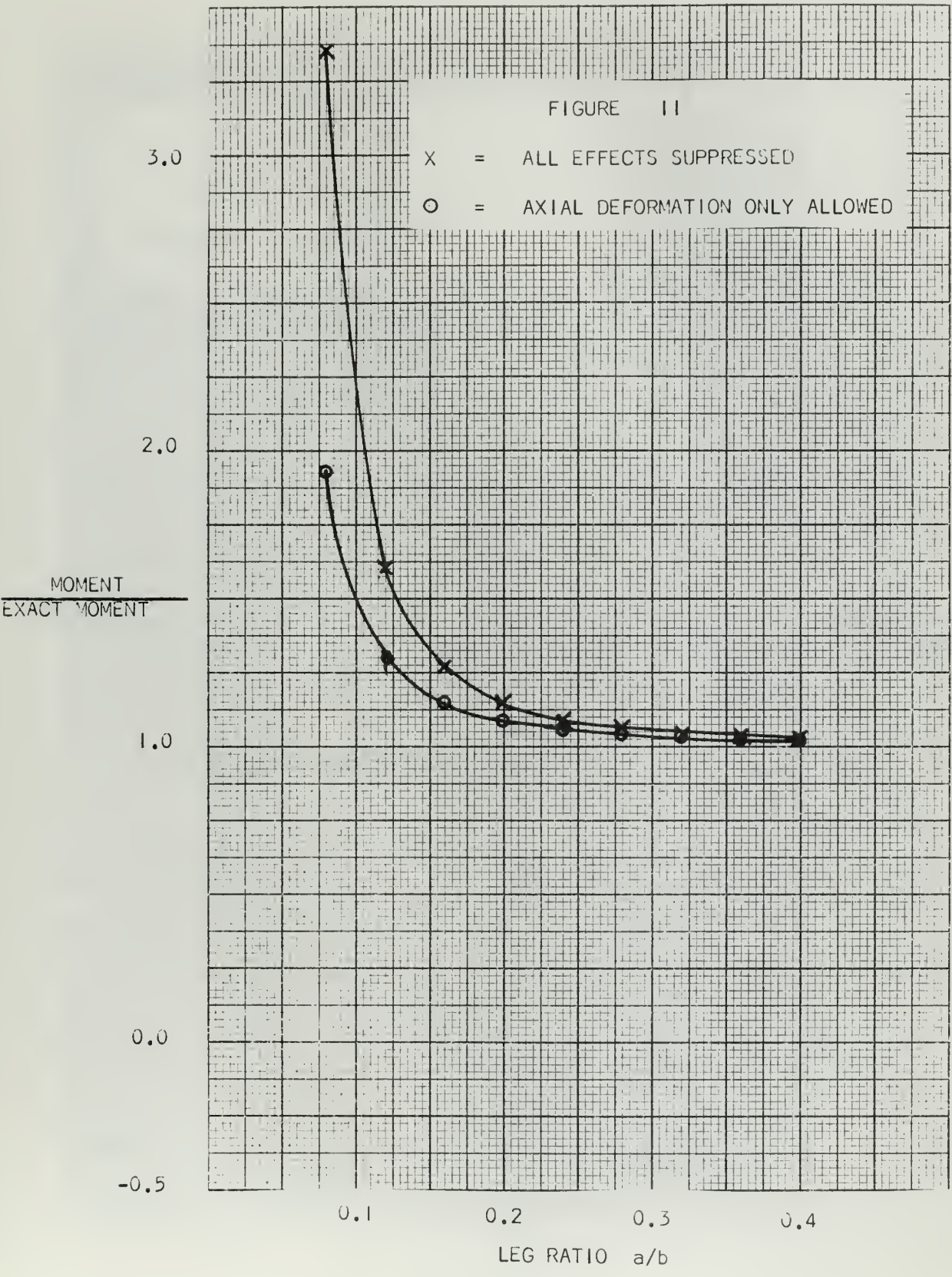


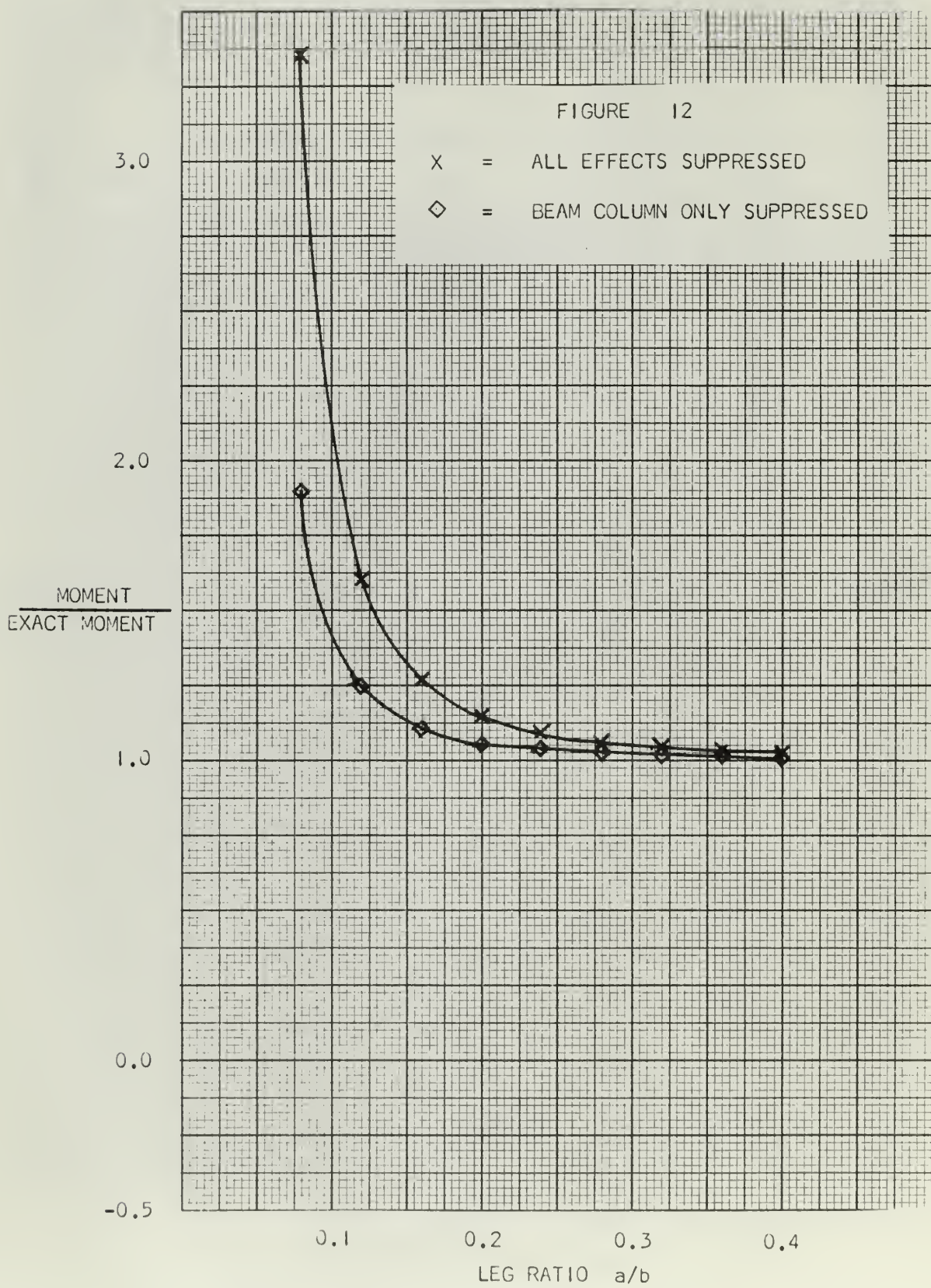


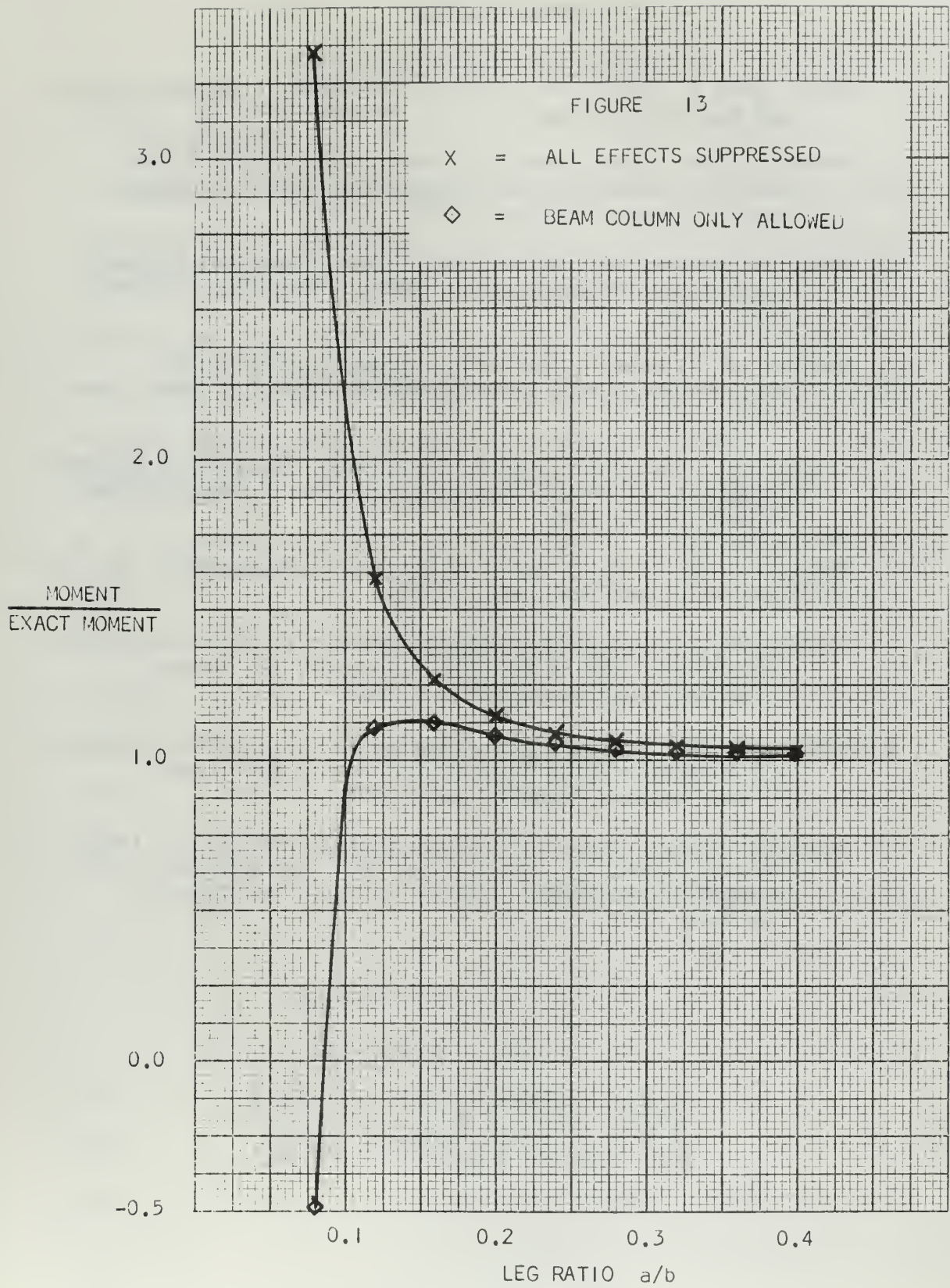












BIBLIOGRAPHY

1. Brock, John E., "Shear Distribution in Piping", Heating, Piping, and Air Conditioning, Vol. 35, January 1963, pp. 141-143.
2. Brock, John E., "Some Secondary Effects In A Simple Piping Structure Under Heating", Journal of Applied Mechanics, Vol. 31, March 1964, pp. 88-90.
3. Dorn, William S., and Daniel D. McCracken, Numerical Methods and Fortran Programming, New York: John Wiley and Sons, Inc., 1964.
4. Goodier, J. N., and S. Timoshenko, Theory of Elasticity, Second Edition, New York: McGraw-Hill Book Company, Inc., 1951.
5. MacCullough, Gleason H., and S. Timoshenko, Elements of Strength of Materials, Second Edition, New York: D. Van Nostrand Company, Inc., 1940.
6. McCracken, Daniel D., A Guide To Fortran IV Programming, New York: John Wiley and Sons, Inc., 1965.
7. Timoshenko, S., Strength of Materials, Part I, Elementary Theory and Problems, Third Edition, Princeton, New Jersey: D. Van Nostrand Company, Inc., 1955.
8. Timoshenko, S., Strength of Materials, Part II, Advanced Theory and Problems, Third Edition, Princeton, New Jersey: D. Van Nostrand Company, Inc., 1956.
9. American Standard Code for Pressure Piping, ASA B31.1 - 1955, UDC-621.64.002.1 .2, Published by: THE AMERICAN SOCIETY OF MECHANICAL ENGINEERS, 29 West 39th St., New York 18, New York, 1955.

APPENDIX A

COMPUTER PROGRAM NOMENCLATURE

A-1 MAIN PROGRAM

NOTE: Where possible the variable names of the computer program are identified by the notation of Tables I and II. Also, in some cases the same physical quantity may be represented in the program by more than one variable name. In such cases the extreme right hand column will indicate the general area of the program where the physical quantity takes on the indicated variable name.

<u>VARIABLE NAME</u>	<u>TABLE I & II NOTATION</u>	<u>COMMENT</u>
A	a	Computation Section
AA(N)	a	Loading Section
A1-A10	$\alpha_1 - \alpha_{10}$	
AB1-AB10	$\bar{\alpha}_1 - \bar{\alpha}_{10}$	
ALIT(N)	e	Loading Section
ALPHA	Subroutine ALPHA	
AREA	A	
B	b	Computation Section
BB(N)	b	Loading Section
B1	$(e+pH)a - \lambda$	
B2	$(e+pH)b - \bar{\lambda}$	
BET1	$[a^2(\alpha_1 + \eta\alpha_4)] / (a\alpha_3 + b\bar{\alpha}_3)$	
BET2	$[b^2(\bar{\alpha}_1 + \bar{\eta}\bar{\alpha}_4)] / (a\alpha_3 + b\bar{\alpha}_3)$	
COEFI	Material coefficient of thermal expansion	
DELT1-DELT4	Size of δ_{par} for each parameter 1-4 respectively	
DET	$g_{11} \times g_{22} - g_{12} \times g_{21}$	
DIN	pipe inside diameter	Computation Section
DOUT	pipe outside diameter	Computation Section

COMPUTER PROGRAM NOMENCLATURE (cont.)

<u>VARIABLE NAME</u>	<u>TABLE I & II NOTATION</u>	<u>COMMENT</u>
DTEMP		Difference of Temperature
E	E	
ELIT	e	Computation Section
EP1,EP2	ϵ_1, ϵ_2	
ETA	η	
ETAB	$\bar{\eta}$	
EXMOM(N)		Exact moment for a particular parameter variable, i.e. with all secondary effects allowed.
EXSCAL		Applicable to graphic output: X scale in units per inch
FAC1-FAC4		Four factors used for either suppressing or allowing their corresponding secondary effects.
FACTOR		Subroutine FACTOR
F1,F2		Assumed values of the forces F_1, F_2 at the beginning of an iteration.
F1CRIT,F2CRIT		Euler Buckling load for F_1 and F_2 .
F1N,F2N		Newly calculated values of F_1 and F_2 from a particular iteration.
FTE11-FTE162		See section on input Data
G	G	Computation Section
g_{ij}		coefficients defined in equations (16)-(19)
H	H	
I		Counter use to count the number of iterations required for convergence.
IA		Factor to aid in determining improved values of F's
LABEL		Argument for Subroutine DRAW
ITITLE		Argument for Subroutine DRAW

COMPUTER PROGRAM NOMENCLATURE (cont.)

<u>VARIABLE NAME</u>	<u>TABLE I & II NOTATION</u>	<u>COMMENT</u>
IXUP	Argument for Subroutine DRAW	
IYRIGH	Argument for Subroutine DRAW	
J	Counter used to count the number of subincrements between the larger main increment.	
KHAT	Secondary Effect combination designator	
KKK	Input variable that assigns value to IA	
KPAR1	Input variable that designates parameter to be investigated	
L	Input variable that aids in determining Δ PAR.	
LABEL	Argument for Subroutine DRAW	
LL1	Input variable that sets size of subincrements, δ par.	
MA	Input variable that designates final value of subscript,N.	
MB	Initial value of MT	
MC	Final value of MT	
MD	Input variable that designates type of output	
MIT	Input variable that specifies the maximum number of iterations that will be performed before terminating operation on a particular set of data.	
MT	Integer variable that specifies the combination of secondary effects to be considered.	
N	Subscript, used to identify calculated values of desired quantities with the corresponding parameter value.	
NN	Initial value of the subscript N	
P	p	Computation Section
PAR2	A designation for the second parameter, (Radius of Gyration)/(Total Length), in the computational part of the program.	

COMPUTER PROGRAM NOMENCLATURE (cont.)

<u>VARIABLE NAME</u>	<u>TABLE I & II NOTATION</u>	<u>COMMENT</u>
PIE	π	
PIESQ	π^2	
POIS	ν	Computation Section
Q	p	Input Data Section
R	a	Input Data Section
RAGY	Radius of Gyration	
RHO	ρ	
S	b	Input Data Section
SFAC	ζ	
W	w	
WBAR	\bar{w}	
XDIN(N)	pipe inside diameter	Loading Section
XDOUT(N)	pipe outside diameter	Loading Section
XLA	λ	
XLAB	$\bar{\lambda}$	
XMU1	μ_1	
XMU2	μ_2	
XN	Moment in Computation Section	
XP(N)	p	Loading Section
YSCALE	Applicable to graphic output: Y scale in units per inch.	
XI	I	
ZIN	pipe inside diameter	Input Data Section
ZLEGR(N)	First parameter, a/b	
ZLEN	Total Length, a + b	
ZMOM	moment	

COMPUTER PROGRAM NOMENCLATURE (cont.)

<u>VARIABLE NAME</u>	<u>TABLE I & II NOTATION</u>	<u>COMMENT</u>
ZOOM	Initial thermal strain	
ZOUT	pipe outside diameter	
ZPAR2(N)	Second parameter, (Radius of Gyration)/(Total Length)	
ZPAR3(N)	Third parameter, Pressure/Modulus of Elasticity	
ZRAT1(N)-ZRAT16(N)	Ratios of the Moments of the sixteen combinations divided by the exact moment for that combination.	
ZRATIO	Same as the ZRAT()'S except this value is not stored.	

A-2 SUBROUTINE ALPHA

<u>VARIABLE NAME</u>	<u>TABLE I & II NOTATION</u>	<u>COMMENT</u>
C	η or $\bar{\eta}$	
EP1	ϵ_1	
T1-T10	$\alpha_1 - \alpha_{10}$ or $\bar{\alpha}_1 - \bar{\alpha}_{10}$	
TX	$ w $ or $ \bar{w} $	
Y	$(-w)^{1/2}$ or $(-\bar{w})^{1/2}$	
Z	$(w)^{1/2}$ or $(\bar{w})^{1/2}$	

A-3 SUBROUTINE FACTOR

<u>VARIABLE NAME</u>	<u>TABLE I & II NOTATION</u>	<u>COMMENT</u>
JT		Integer variable, MT
LLAT		Secondary effect combination designator
R1-R4		Four factors used for either suppressing or allowing their corresponding secondary effects.

A-4 SUBROUTINE DRAW-DESCRIPTION FROM SUBROUTINE LIBRARY

A. IDENTIFICATION:

TITLE: General Graph Output Subroutine
SUBROUTINE NAME: DRAW(J7-NPS-DRAW for CDC 1604 revised for IBM 360)
DATE: July, 1967
PROGRAMMER: Revised by Patricia Johnson

B. PURPOSE:

This subroutine generates data for plotting graphs on a CalComp plotter. Provision is made for curve drawing and point plotting, automatic scaling, graph titling and axis annotating. An attempt was made to provide a considerable amount of flexibility, at the expense, necessarily, of a relatively large number of arguments and a rather high memory requirement.

C. USAGE:

1. Definitions:

In what follows the word "graph" will be taken to mean one piece or frame of graph paper on which there may be plotted one or more curves and/or sets of points. A "curve" will mean a continuous line generated by the sequence of straight lines joining successive points of the set defining the curve. A "point plot" will describe the representation of a succession of points by means of symbols (such as a cross) on the graph. The points are not connected in a point plot.

2. Calling Arguments:

All necessary information is transferred to DRAW through the calling arguments. The call statement is: CALL DRAW (NUMPTS, X,Y,MODCUR,ITYPE,LABEL,ITITLE,EXSCAL,YSCALE,IXUP,IYRIGH, MODXAX,MODYAX,IWIDE,IHIGH,IGRID,LAST)

It is important to realize that one and only one curve or set of points is plotted each time DRAW is called. However, it is possible to call DRAW repeatedly if several curves and/or sets of points are wanted on one graph.

The calling arguments are as follows:

- a. NUMPTS: The number of points defining a curve ($2 \leq \text{NUMPTS} \leq 900$), of the number of points to be point plotted ($2 \leq \text{NUMPTS} \leq 30$).
- b. X: The array of X-ordinates, X, must be dimensioned at least equal to NUMPTS in the calling program.
- c. Y: The array of Y-ordinates, Y, must be dimensioned at least equal to NUMPTS in the calling program.
- d. MODCUR: Controls the number of curves, and/or sets of points on one graph:
 - = 0 This is the only curve, or set of points, to be plotted on this graph.
 - = 1 This is the first of two or more curves, and/or sets of points, to be plotted on this graph.
 - = 2 This is an intermediate curve, or set of points.
 - = 3 This is the last curve, or set of points, for this graph.
- e. ITYPE: Controls the type of plot (i.e., curve or point plot):
 - = 0 This set of points is to be represented by a curve.
 - = 1 These points are to be plotted with a cross (x).
 - = 2 These points are to be plotted with a plus (+).
 - = 3 These points are to be plotted with a square (□).
 - = 4 These points are to be plotted with a diamond (◇).
 - = 5 These points are to be plotted with a triangle (△).

f. LABEL: This is a Hollerith curve or point identifier.

If a curve is being drawn, LABEL must have 4 characters (including any blanks), and these will be reproduced beside the end of the curve. This argument can be set in the calling program by a declaration statement, e.g.,

```
REAL LABEL/4H1234/
```

or

```
REAL LABEL/'1234'/
```

or

```
REAL LABEL/4H
```

 / (The latter must be used when no label is wanted.)

If a set of points is being plotted, LABEL is an 8-character identifier. The first 4 characters will be reproduced beside the first point, and the last four characters will be reproduced alongside the last point. This argument can be set by a declaration statement, e.g.,

```
REAL*8 LABEL/8HFRSTLAST/
```

or

```
REAL LABEL(2)/4HFRST,4HLAST/
```

or

```
REAL*8 LABEL/8H
```

 / (The latter must be used when no label is wanted).

The above arguments, a. through f. (and q.), have meaning every time DRAW is called. On the other hand, the remaining arguments, g. through p., have no meaning except when MODCUR = 0 or 1.

- g. ITITLE: An array of 96 EBCDIC characters, the first 48 of which will form the first title line, and the last 48 the second. The array must be dimensioned in the calling program, must contain the user's job identification, and must have unwanted characters set to blank. This may be done with a declaration statement such as:
- ```
REAL*8 ITITLE(12) / '96EBCDIC character' /
```
- h. EXSCAL: X-scale in units per inch. EXSCAL will always be rounded off to one figure significance. If EXSCAL = 0, the X-scale will be computed by DRAW. This is called auto-scale.
- i. YSCALE: Y-scale in units per inch, with properties corresponding to those of EXSCAL.
- j. IXUP: Distance, in inches, of the X-axis from the bottom of the graph ( $0 \leq IXUP \leq IHIGH$ ). This argument will be ignored unless MODXAX = 2, see below.
- k. IYRIGH: Distance, in inches, of the Y-axis from the left of the graph ( $0 \leq IYRIGH \leq IWIDE$ ). This will be ignored unless MODYAX = 2, see below.
- l. MODXAX: Determines the mode of the X-axis location:
- = 0 The X-axis will be located automatically by DRAW, with the origin of Y on the graph.
  - = 1 The X-axis will be automatically located by DRAW, with the origin of Y moved (in one's imagination) an integer number of inches above or below the graph, if this is appropriate. This option can be used only if the Y-scaling is automatic (YSCALE = 0).

- = 2            The X-axis location will be as specified by LXUP.
- m. MODYAX:    Determines the mode of Y-axis location in the same way as MODXAX, above, governs the X-axis location.
- n. IWIDE:      Width of graph in inches ( $1 \leq IWIDE \leq 9$ ). If IWIDE is out of this range, a value of 8 will be assumed.
- o. IHIGH:      Height of graph in inches ( $1 \leq IHIGH \leq 15$ ). If IHIGH is out of range, a value of 8 will be assumed.
- p. IGRID:      If IGRID = 1, a 1" x 1" grid will be superimposed on the graph. This is useful only if plain paper is used on the CalComp Plotter.
- q. LAST:       Indicates to the calling program whether the previous plot was completed successfully. The codes are:
- = 0            Last plot was completed successfully
- = 1            Last plot was not completed successfully.
- = 2            Last plot was not completed successfully and no further graph output will be attempted until DRAW is next entered with MODCUR = 1 or 0
- = 3            An attempt was made to enter DRAW with MODCUR  $\neq$  1 or 0 while the error lockout was set. This argument must always be a variable name and never a number in the call statement.



### 3. Notes and Comments:

- a. The graph scales and, if  $\text{MODXAX} = 1$  and/or  $\text{MODYAX} = 1$ , the amounts of origin offset are always output as part of the graph title.
- b. There are internal checks of the input to DRAW to prevent incorrect use. If an input error is detected, an attempt will be made, where possible, to complete the plot. If an argument is "corrected" in this process, the user will be so informed on the standard output. If it is not possible to complete the plot, the user will be informed of the reason by a message on the standard output.
- c. If part or all of a curve would fall more than 0.6" laterally beyond the ends of the X-axis, or 0.7" vertically beyond the ends of the Y-axis, the X and/or Y ordinates will be limited so that the curve will typically become a line along part or all of the boundary of the graph as here defined.
- d. If one or more points of a point plot would fall outside the graph area, the plot of the point, or points will be inhibited. The number of such points will be reported to the user on the standard output.
- e. It should be pointed out that the X and Y scaling and axis locating processes are entirely independent, so that, for example, X might be auto-scaled, while the Y-scale is specified. At the same time the X-axis might be located automatically, while the Y-axis location is specified.
- f. It must be remembered that the scales and axis locations of a multi-plot graph are set when DRAW is called for the first time (with  $\text{MODCUR} = 1$ ). Thus the user must attempt, at the



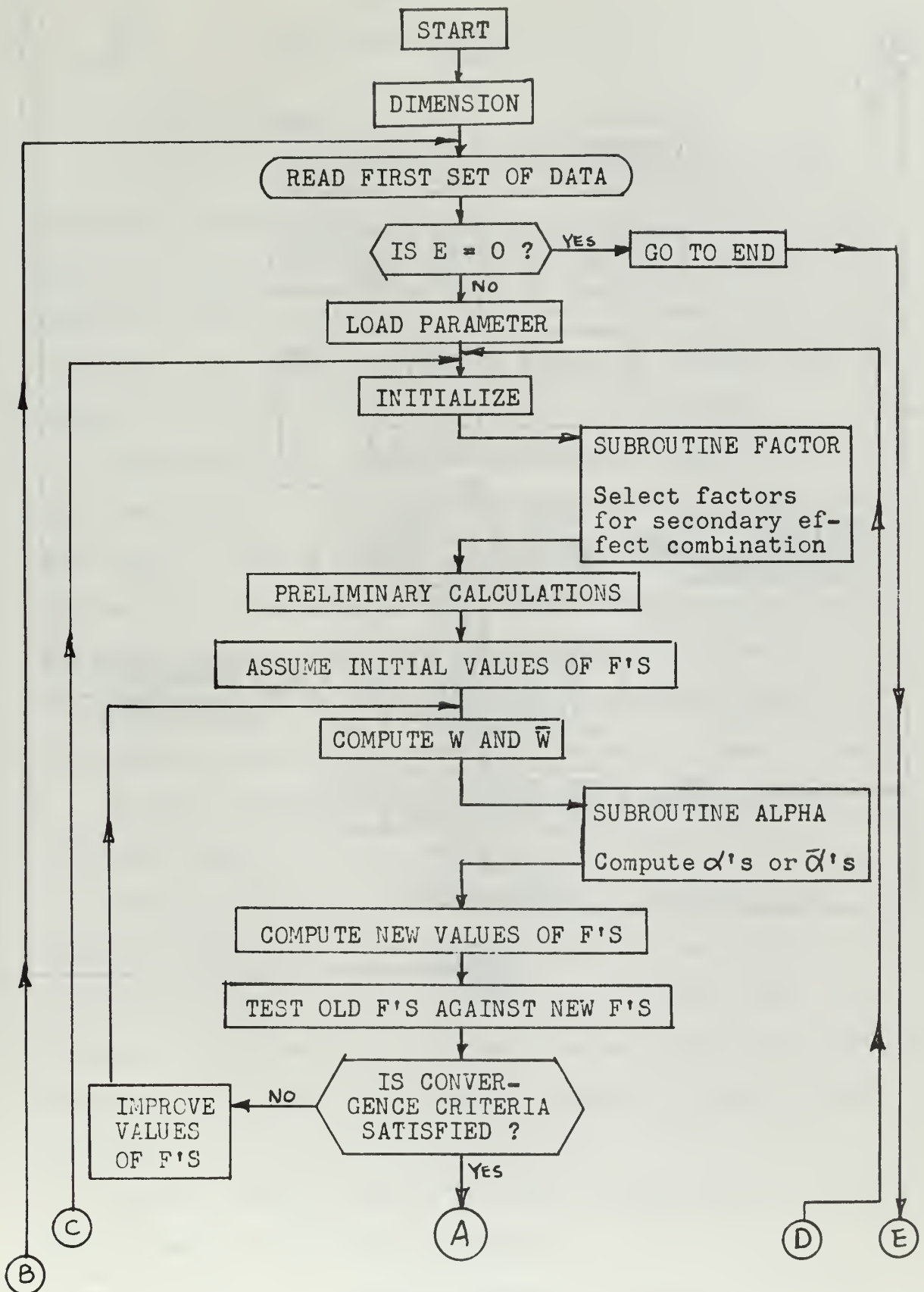
time to achieve scaling and axis location which will be appropriate to all the plots he intends to make on the one graph. Particularly if the automatic features of DRAW are selected, foresight will be demanded of the user in this respect.

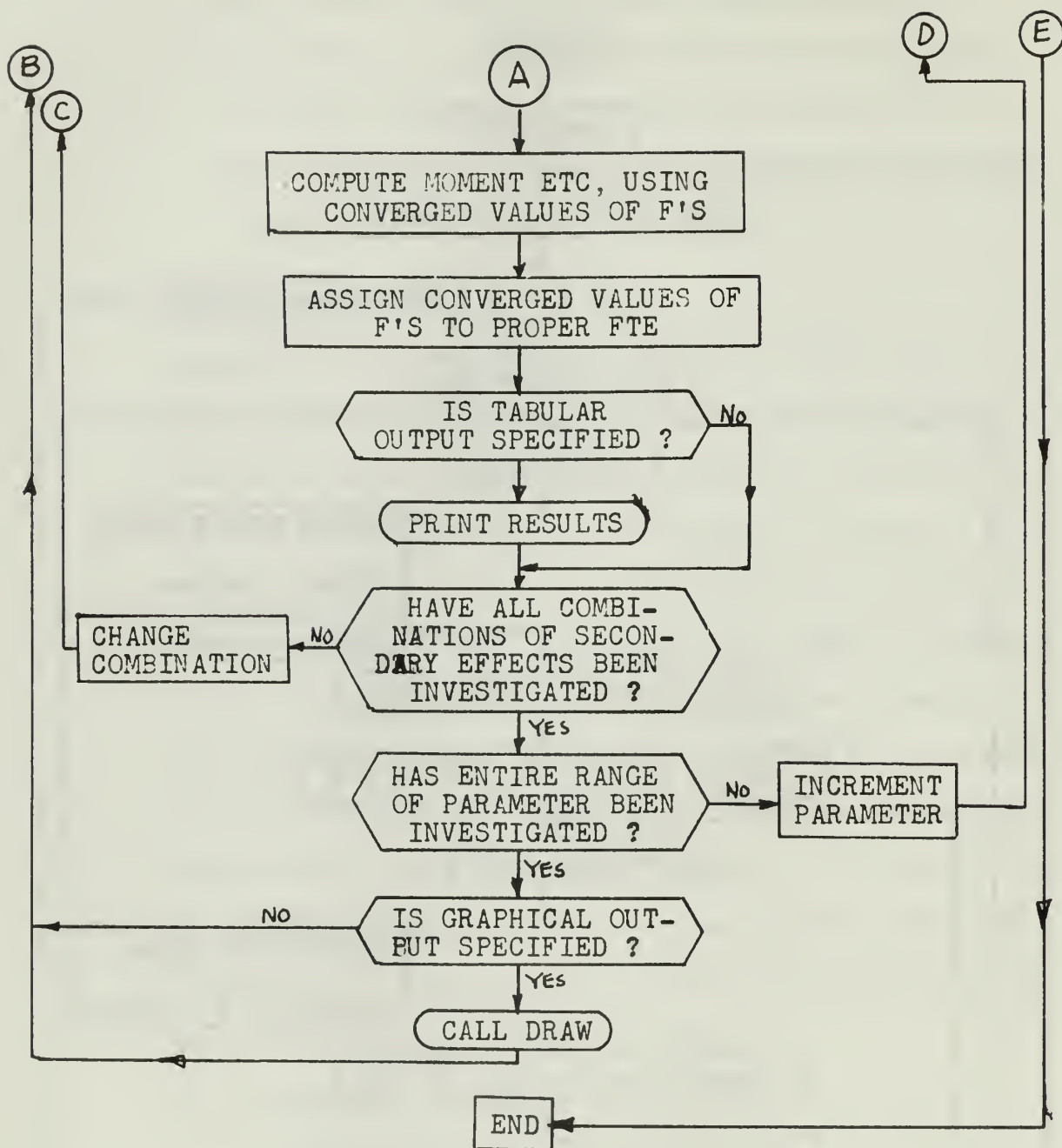
#### 4. Auto-Scale Properties:

The scale factor is chosen from amongst the values 1, 2, or 5 units per inch, or some power of 10 times one of those values. A curve, or set of points which is plotted with auto-scale will normally lie entirely within the graph area is defined in 3.d., above. The only exception may occur if an axis is placed, by the user, along one edge of the graph (e.g.,  $IXUP = 0$ ,  $MODXAX = 2$ ). In such a case, points "outside" the axis are not considered in the selection of a scale factor (e.g., negative  $X_1$  do not affect the choice of scale when  $IXUP = 0$ ). If automatic axis location as well as auto-scale is selected, the plot, if it does not fill the graph area, will be placed as far as possible towards the bottom-left of the graph area consistent with the fact that the axes can be set in increments of 1" only.

1st revision - 31 July 1967.

APPENDIX B  
COMPUTER PROGRAM FLOW CHART





## APPENDIX C

### DATA CARD DESCRIPTION

#### C-1 Data Card Placement

For one set of data, six data cards are required to give the program the necessary input. The first card gives the physical data such as leg length, pipe size and material, and other quantities that describe the physical situation. The second card supplies the control variables to the program. These control variables indicate to the program what parameter is to be investigated and its range.

The last four cards supply the initial values given to the FTE's (see page 26). Each of the last four cards contain eight of these values. Unless a particular computer run starts near the critical part of the parameter range (where convergence is difficult) the values on the last four cards are usually zero. If such is the case, blank cards may be used; however, when values other than zero are supplied as input, the format shown in Appendix C-2 must be used.

It must be remembered that six data cards must always be used for a particular set of input whether or not the FTE's are zero.

Any number of sets of six data cards can be supplied to the program for any one run. Consideration should be given to time requirements, however. After the final set of six data cards, one blank card follows and then, after that, the last control card (see Appendix D). The blank card is used to terminate the run by telling the program that all data has been considered.

The user should be familiar with fixed and floating point variables and the common Fortran field specifications.<sup>3</sup>

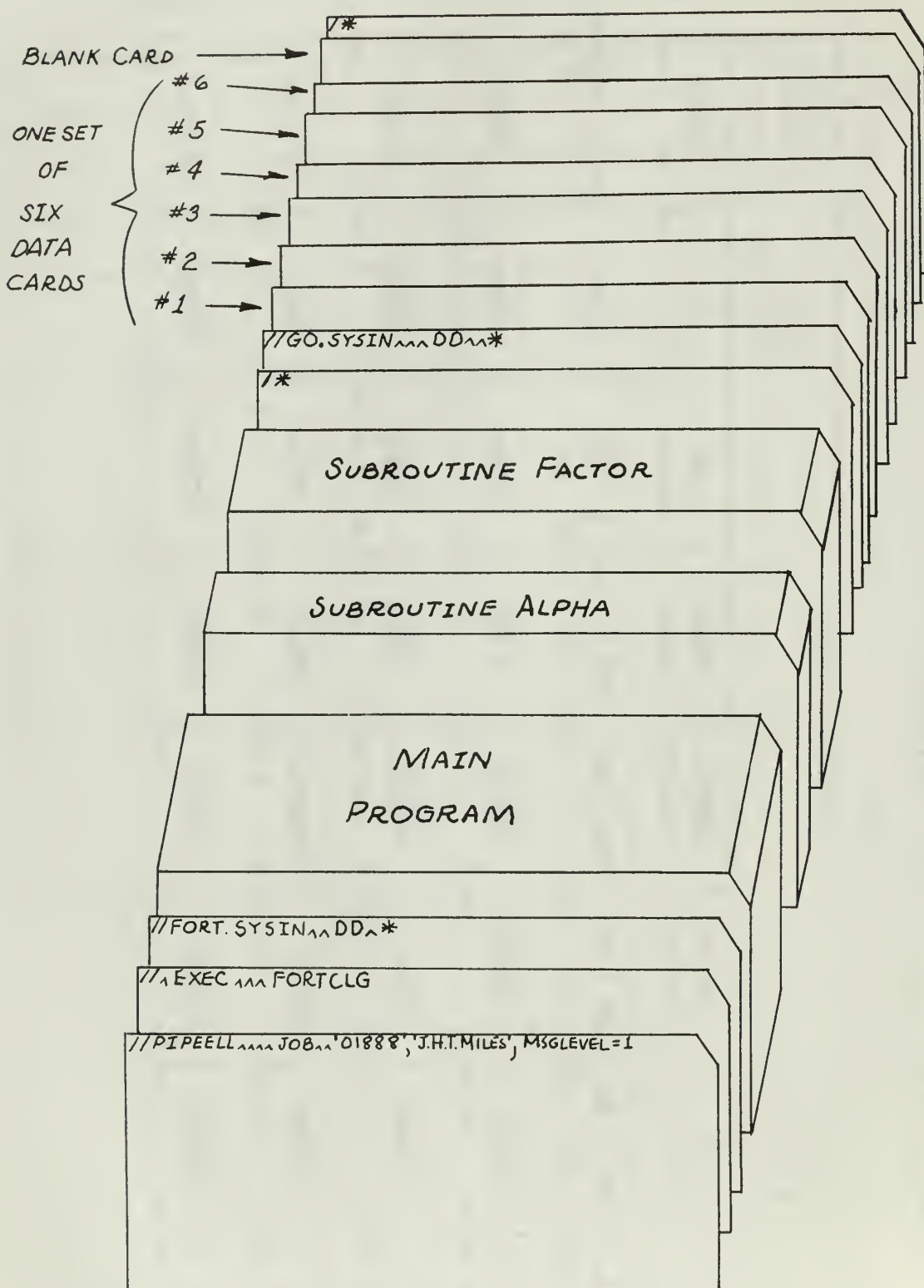
-----  
<sup>3</sup> Daniel D. McCracken, A Guide to Fortran IV Programming (New York: John Wiley and Sons, Inc., 1965), pp. 3-5, 13, 84.

Appendix C-2 shows a typical input (all six data cards). The extreme left hand column indicates the data card number (this order must always be followed). For each data card there are two rows shown. The top row indicates the variable name whose value is shown directly below in the next row. This second row is the actual value of the variable named above it and appears just as the data card itself does. The top row of the two in each case serves only to aid in identification of the variable.



| PROGRAMMER                    |        | Miles, J.H.T.           |        | PROGRAM                                    |        | PIPEELL                |        | DATE                                                     |   |    |      |      |     |      |        |
|-------------------------------|--------|-------------------------|--------|--------------------------------------------|--------|------------------------|--------|----------------------------------------------------------|---|----|------|------|-----|------|--------|
| PAGE                          |        | OF                      |        | SPECIAL INSTRUCTIONS                       |        | DATE COMPLETED         |        | SUBMIT FOR PROCESSING                                    |   |    |      |      |     |      |        |
| CARD PUNCH OPERATOR           |        |                         |        |                                            |        |                        |        | <input type="checkbox"/> YES <input type="checkbox"/> NO |   |    |      |      |     |      |        |
| STATE-6<br>P<br>MENT<br>E NO. |        | O = ZERO<br>Ø = ALPHA 0 |        | FORTAN STATEMENT<br>I - ONE<br>I = ALPHA I |        | 2 = TWO<br>Z = ALPHA Z |        | SERIAL<br>NUMBER                                         |   |    |      |      |     |      |        |
| 1                             | 360.0  | R                       | S      | ZIN                                        | MD     | MIT                    | KKK    | LL1                                                      | Q | E  | 274  | 0.08 | 723 | 0-05 | COEPI  |
| 2                             | 1      | 25                      | 1      | 26                                         | 1      | 16                     | 1      | 100                                                      | 6 | 40 | 0.01 | 1.2  |     |      | YSCALE |
| 3                             | FTE11  | FTE12                   | FTE21  | FTE22                                      | FTE31  | FTE32                  | FTE41  | FTE42                                                    |   |    |      |      |     |      |        |
| 4                             | FTE51  | FTE52                   | FTE61  | FTE62                                      | FTE71  | FTE72                  | FTE81  | FTE82                                                    |   |    |      |      |     |      |        |
| 5                             | FTE91  | FTE92                   | FTE101 | FTE102                                     | FTE111 | FTE112                 | FTE121 | FTE122                                                   |   |    |      |      |     |      |        |
| 6                             | FTE131 | FTE132                  | FTE141 | FTE142                                     | FTE151 | FTE152                 | FTE161 | FTE162                                                   |   |    |      |      |     |      |        |
| 7                             | 1      | 2                       | 3      | 4                                          | 5      | 6                      | 7      | 8                                                        | 9 | 10 | 11   | 12   | 13  | 14   | 15     |





APPENDIX D  
CONTROL CARD ARRANGEMENT

## E-1 Main Program

UUUUUU

```

732 XDIN(N)=ZIN
 XDOUT(N)=ZOUT
 ALIT(N)=CCEFI*DTEMP
 IF(N-MA)732,733,733
 N=N+1
 GO TO 734
733 GO TO 706
735 IF(KPAR1-2)209,752,740
752 V=1
 XDIN(1)=ZIN
 XDOUT(N)=ZOUT
 XDIN(N+1)=XDIN(N)+7IN/L
 XP(N)=Q
 ALIT(N)=CCEFI*DTEMP
 AA(N)=R
 BB(N)=S
 IF(N-MA)754,755,755
754 N=N+1
 GO TO 753
755 GO TO 706
740 IF(KPAR1-3)209,741,746
741 N=1
 XP(1)=Q
 XP(N+1)=XP(N)+Q/L
 AA(N)=R
 BB(N)=S
 XDIN(N)=ZIN
 XDOUT(N)=ZOUT
 ALIT(N)=CCEFI*DTEMP
 IF(N-MA)743,745,745
743 N=N+1
 GO TO 744
745 GO TO 706
746 IF(KPAR1-4)209,747,209
747 N=1
 ZOOM=CCEFI*DTEMP
 ALIT(1)=ZOOM
 ALIT(N+1)=ALIT(N)+ZOOM/L
 XP(N)=Q
 XDIN(N)=ZIN
 XDOUT(N)=ZOUT
 AA(N)=R
 BB(N)=S
 IF(N-MA)749,750,750
749 N=N+1
 GO TO 748
750 GO TO 706
706 N=NN

```

```

00000510
00000520
00000530
00000540
00000550
00000560
00000570
00000580
00000590
00000600
00000610
00000620
00000630
00000640
00000650
00000660
00000670
00000680
00000690
00000700
00000710
00000720
00000730
00000740
00000750
00000760
00000770
00000780
00000790
00000800
00000810
00000820
00000830
00000840
00000850
00000860
00000870
00000880
00000890
00000900
00000910
00000920
00000930
00000940
00000950
00000960
00000970
00000980

```

```

303 CONTINUE
 CLUCK = ITIME(C)*.01
 IF(N.NF.NN.AND.MD.EQ.2)GO TO 305
 WRITE(6,301)
301 FORMAT(1H1,18X,1HA,9X,1HB,8X,3HDIN,7X,4HDOUT,5X,5HTEMP,7X,1HP,9X,
1HE,8X,5HCOEF1,8X,5HKPAR1//)
302 WRITE(6,302)K,S,ZIN,ZOUT,DTEMP,Q,F,COEF1,KPAR1
305 FORMAT(14X,7D10.2,D10.3,4X,I5//)
 CONTINUE
 IF(MD.EQ.2)GO TO 306
 WRITE(6,217)
217 FORMAT(14X,1HN,4X,2HMT,3X,1CHDPSIGNATOR,4X,5HMOVMENT,7X,8HMMRATIO,
19X,3HF1N,12X,3HF2N,8X,15HPARAMETER VALUE,8X,1CHITERATIONS//)
C
C
C DIVIDE PARAMETER VALUES INTO SUB-INCREMENTS.
306 DELT1=(AA(N+1) - AA(N))/LL1
 DELT2 = (XCIN(N + 1) - XCIN(N))/LL1
 DELT3 = (XP(N+1) - XP(N))/LL1
 DELT4 = (ALIT(N+1) - ALIT(N))/LL1
 A=AA(N)
 B=BB(N)
 ELIT = ALIT(N)
 DIN = XDIN(N)
 DCUT = XDOUT(N)
 P = XP(N)
 J=1
650 MT = MB
C
C
C SELECT FACTORS THAT PROVIDE DESIRED COMBINATION OF SECONDARY
 EFFECTS.
709 CALL FACTOR(MT,FAC1,FAC2,FAC3,FAC4,KHAT)
C
C
C PRELIMINARY CALCULATIONS MADE FROM INPUT DATA.
 PCIS=0.3
 RHO=DIN/DCUT
 H = (1.0 - 2.0*PCIS)*(RHO**2.0)/(E*(1.0 - (RHO**2.0)))
 XMU1=(7.0+14.0*PCIS+8.0*(PCIS**2.0))/(6.0*((1.0+POIS)**2.0))
 XMU2=(10.0+20.0*POIS+8.0*(PCIS**2.0))/(3.0*((1.0+POIS)**2.0))
 SFAC=(XMU1+(XMU2*(RHO**2.0))/(1.0+RHO**2.0)**2.0)*FAC2
 AREA=.785398*(DCUT**2.0-DIN**2.0)
 ZI=.049087*(DCUT**4.0-DIN**4.0)
 G=F/(2.0*(1.0+POIS))
 RAGY = DSQRT(ZI/AREA)
 ZLEN = A + B
 PAR2 = RAGY/7LEN

```

```

ETA=(SFAC*E*Z1)/(AREA*G*(A **2.0))
ETAR=(SFAC*F*Z1)/(AREA*G*(R **2.0))
EPI = .01

```

```

SELECT FIRST APPROXIMATIONS FOR F1 AND F2 FROM LAST CONVERGED
VALUES OF THE CORRESPONDING COMBINATION OF SECONDARY EFFECTS.

```

C  
C  
C  
C

```

IF(MT.NE.1)GC TO 625
F1=FTE11
F2 = FTE12
GC TO 100
625 CONTINUE
IF(MT.NE.2)GC TO 626
F1=FTE21
F2=FTE22
GC TO 100
626 CONTINUE
IF(MT.NE.3)GC TO 627
F1=FTE31
F2=FTE32
GC TO 100
627 CONTINUE
IF(MT.NE.4)GC TO 628
F1=FTE41
F2=FTE42
GC TO 100
628 CONTINUE
IF(MT.NE.5)GC TO 629
F1=FTE51
F2=FTE52
GC TO 100
629 CONTINUE
IF(MT.NE.6)GC TO 630
F1 = FTE61
F2 = FTE62
GC TO 100
630 CONTINUE
IF(MT.NE.7)GC TO 631
F1=FTE71
F2=FTE72
GC TO 100
631 CONTINUE
IF(MT.NE.8)GC TO 632
F1=FTE81
F2=FTE82
GC TO 100
632 IF(MT.NE.9)GC TO 633

```

```

00001470
00001480
00001490
00001500
00001510
00001520
00001530
00001540
00001550
00001560
00001570
00001580
00001590
00001600
00001610
00001620
00001630
00001640
00001650
00001660
00001670
00001680
00001690
00001700
00001710
00001720
00001730
00001740
00001750
00001760
00001770
00001780
00001790
00001800
00001810
00001820
00001830
00001840
00001850
00001860
00001870
00001880
00001890
00001900
00001910
00001920
00001930
00001940

```







[illegible]

```

C
C
C
C
C
207 ZMOM= BEI1 * F1N + BEI2 * F2N
 IF PARAMETER VALUE IS EQUAL TO THAT OF A STORED VALUE, I.E., INDEX
 N EQUALS AN INTEGER VALUE. COMPUTE OTHER NECESSARY VALUES FOR
 STORAGE.
 IF (J.EQ.1.AND.MT.EQ.1) GO TO 710
 GO TO 601
710 EXMOM(N)=ZMOM
 ZRATIO = ZMOM/EXMOM(N)
 ZRAT1(N) = ZMOM/EXMOM(N)
 GO TO 602
601 CONTINUE
602 IF (J.NF.1) GO TO 603
 PIF = 3.1415927
 PIESQ = PIF**2
 F1CRIT = (PIESQ*E*ZI)/B**2
 F2CRIT = (PIESQ*E*ZI)/A**2
 ZLEGR(N) = AA(N)/RB(N)
 ZPAR2(N) = PAR2
 ZPAR3(N) = XP(N)/E
 ASSIGN NEWLY CONVERGED VALUES OF F1 AND F2 TO PROPER FTE.
C
C
C
680 IF (MT.NE.2) GO TO 68C
 ZRAT2(N) = ZMOM/EXMOM(N)
 ZRATIO = ZMOM/EXMOM(N)
 GO TO 603
 CONTINUE
681 IF (MT.NE.3) GO TO 681
 ZRAT3(N) = ZMOM/EXMOM(N)
 ZRATIO = ZMOM/EXMOM(N)
 GO TO 603
 CONTINUE
682 IF (MT.NE.4) GO TO 682
 ZRAT4(N) = ZMOM/EXMOM(N)
 ZRATIO = ZMOM/EXMOM(N)
 GO TO 603
 CONTINUE
683 IF (MT.NE.5) GO TO 683
 ZRAT5(N) = ZMOM/EXMOM(N)
 ZRATIO = ZMOM/EXMOM(N)
 GO TO 603
 CONTINUE
 IF (MT.NE.6) GO TO 684
 ZRAT6(N) = ZMOM/EXMOM(N)
 ZRATIO = ZMOM/EXMOM(N)

```

00002910  
00002920  
00002930  
00002940  
00002950  
00002960  
00002970  
00002980  
00002990  
00003000  
00003010  
00003020  
00003030  
00003040  
00003050  
00003060  
00003070  
00003080  
00003090  
00003100  
00003110  
00003120  
00003130  
00003140  
00003150  
00003160  
00003170  
00003180  
00003190  
00003200  
00003210  
00003220  
00003230  
00003240  
00003250  
00003260  
00003270  
00003280  
00003290  
00003300  
00003310  
00003320  
00003330  
00003340  
00003350  
00003360  
00003370  
00003380

|     |                                                                                                                              |                                                                                  |
|-----|------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------|
| 684 | GG TO 603<br>CONTINUE<br>IF(MT.NE.7)GC TO 685<br>ZRAT7(N) = ZMOM/EXMOM(N)<br>ZRATIO = ZMCM/EXMOM(N)<br>GO TO 603<br>CONTINUE | 00003390<br>00003400<br>00003410<br>00003420<br>00003430<br>00003440<br>00003450 |
| 685 | IF(MT.NE.8)GC TO 686<br>ZRAT8(N) = ZMCM/EXMOM(N)<br>ZRATIO = ZMCM/EXMOM(N)<br>GO TO 603<br>CONTINUE                          | 00003460<br>00003470<br>00003480<br>00003490<br>00003500                         |
| 686 | IF(MT.NE.9)GC TO 687<br>ZRAT9(N) = ZMCM/EXMOM(N)<br>ZRATIO = ZMCM/EXMOM(N)<br>GO TO 603<br>CONTINUE                          | 00003510<br>00003520<br>00003530<br>00003540<br>00003550                         |
| 687 | IF(MT.NE.10)GC TO 688<br>ZRAT10(N) = ZMOM/EXMCM(N)<br>ZRATIO = ZMOM/EXMOM(N)<br>GO TO 603<br>CONTINUE                        | 00003560<br>00003570<br>00003580<br>00003590<br>00003600                         |
| 688 | IF(MT.NE.11)GC TO 689<br>ZRAT11(N) = ZMOM/EXMCM(N)<br>ZRATIO = ZMOM/EXMOM(N)<br>GO TO 603<br>CONTINUE                        | 00003610<br>00003620<br>00003630<br>00003640<br>00003650                         |
| 689 | IF(MT.NE.12)GC TO 690<br>ZRAT12(N) = ZMOM/EXMCM(N)<br>ZRATIO = ZMOM/EXMOM(N)<br>GO TO 603<br>CONTINUE                        | 00003660<br>00003670<br>00003680<br>00003690<br>00003700                         |
| 690 | IF(MT.NE.13)GC TO 691<br>ZRAT13(N) = ZMOM/EXMCM(N)<br>ZRATIO = ZMOM/EXMOM(N)<br>GO TO 603<br>CONTINUE                        | 00003710<br>00003720<br>00003730<br>00003740<br>00003750                         |
| 691 | IF(MT.NE.14)GC TO 692<br>ZRAT14(N) = ZMOM/EXMCM(N)<br>ZRATIO = ZMOM/EXMOM(N)<br>GO TO 603<br>CONTINUE                        | 00003760<br>00003770<br>00003780<br>00003790<br>00003800                         |
| 692 | IF(MT.NE.15)GC TO 693<br>ZRAT15(N) = ZMOM/EXMCM(N)<br>ZRATIO = ZMOM/EXMOM(N)<br>GO TO 603<br>CONTINUE                        | 00003810<br>00003820<br>00003830<br>00003840<br>00003850                         |
| 693 | IF(MT.NE.16)GC TO 603                                                                                                        | 00003860                                                                         |

```

7)RATIO = ZMCM/EXMCM(N)
CONTINUE
IF(MT.NE.1)GC TO 604
FTE11=FIN
FTE12=F2N
GO TO 651
604 CONTINUE
IF(MT.NE.2)GC TO 610
FTE21 = FIN
FTE22 = F2N
GO TO 651
610 CONTINUE
IF(MT.NE.3)GC TO 611
FTE31 = FIN
FTE32 = F2N
GO TO 651
611 CONTINUE
IF(MT.NE.4) GO TO 612
FTE41=FIN
FTE42=F2N
GO TO 651
612 CONTINUE
IF(MT.NE.5)GC TO 613
FTE51=FIN
FTE52=F2N
GO TO 651
613 CONTINUE
IF(MT.NE.6)GC TO 614
FTE61=FIN
FTE62 = F2N
GO TO 651
614 CONTINUE
IF(MT.NE.7)GC TO 615
FTE71 = FIN
FTE72 = F2N
GO TO 651
615 CONTINUE
IF(MT.NE.8)GC TO 616
FTE81=FIN
FTE82=F2N
GO TO 651
616 CONTINUE
IF(MT.NE.9)GC TO 617
FTE91 = FIN
FTE92 = F2N
GO TO 651
617 CONTINUE

```

```

0003870
0003880
0003890
0003900
0003910
0003920
0003930
0003940
0003950
0003960
0003970
0003980
0003990
0004000
0004010
0004020
0004030
0004040
0004050
0004060
0004070
0004080
0004090
0004100
0004110
0004120
0004130
0004140
0004150
0004160
0004170
0004180
0004190
0004200
0004210
0004220
0004230
0004240
0004250
0004260
0004270
0004280
0004290
0004300
0004310
0004320
0004330
0004340

```

```

IF(MT.NE.10)GO TO 618
FTE101 = F1N
FTE102 = F2N
GO TO 651
618 CONTINUE
IF(MT.NE.11)GO TO 619
FTE111 = F1N
FTE112 = F2N
GO TO 651
619 CONTINUE
IF(MT.NE.12)GO TO 620
FTE121 = F1N
FTE122 = F2N
GO TO 651
620 CONTINUE
IF(MT.NE.13)GO TO 621
FTE131 = F1N
FTE132 = F2N
GO TO 651
621 CONTINUE
IF(MT.NE.14)GO TO 622
FTE141 = F1N
FTE142 = F2N
GO TO 651
622 CONTINUE
IF(MT.NE.15)GO TO 623
FTE151 = F1N
FTE152 = F2N
GO TO 651
623 FTE161 = F1N
FTE162 = F2N
651 CONTINUE
IF(J.EQ.1.AND.MD.EQ.1)GO TO 655
IF(J.EQ.1.AND.MD.EQ.3)GO TO 655
GO TO 665
C
C
C
IF PARAMETER VALUE IS A STORED VALUE PRINT RESULTS IF TABULAR
OUTPUT IS SPECIFIED.
655 IF(KPAR1.NE.1)GO TO 656
WRITE(6,657)N,MT,KHAT,7MOM,ZRATIO,F1N,F2N,ZLEGR(N),I
657 FORMAT(11X,I4,2X,I4,7X,I5,2X,D12.5,4D15.5,10X,I5/)
IF(MT.NE.MC)GO TO 665
WRITE(6,658)
658 FORMAT(/15X,39HPARAMETER INVESTIGATED = LEG RATIO A/R/)
GO TO 665
656 CONTINUE
IF(KPAR1.NE.2)GO TO 659

```

```

00004350
00004360
00004370
00004380
00004390
00004400
00004410
00004420
00004430
00004440
00004450
00004460
00004470
00004480
00004490
00004500
00004510
00004520
00004530
00004540
00004550
00004560
00004570
00004580
00004590
00004600
00004610
00004620
00004630
00004640
00004650
00004660
00004670
00004680
00004690
00004700
00004710
00004720
00004730
00004740
00004750
00004760
00004770
00004780
00004790
00004800
00004810
00004820

```



```

660 WRITE(6,660)N,MT,KHAT,ZMOM,ZRATIO,F1N,F2N,ZPAR2(N),I
 FORMAT(11X,I4,2X,I4,7X,I5,2X,D12.5,4D15.5,10X,I5//)
 IF(MT.NE.MC)GO TO 665
661 WRITE(6,661)F1CRIT,F2CRIT
 FORMAT(//15X,57HPARAMETER INVESTIGATED = RADIUS OF GYRATION/TOTAL
1 LENGTH,//15X,15HF1CRITICAL = ,D15.5,10X,15HF2CRITICAL = ,D15.
25//)
 GO TO 665
659 CONTINUE
 IF(KPAR1.NE.3)GO TO 662
663 WRITE(6,663)N,MT,KHAT,ZMOM,ZRATIO,F1N,F2N,ZPAR3(N),I
 FORMAT(11X,I4,2X,I4,7X,I5,2X,D12.5,4D15.5,10X,I5//)
 IF(MT.NE.MC)GO TO 665
664 WRITE(6,664)F1CRIT,F2CRIT
 FORMAT(//15X,66HPARAMETER INVESTIGATED = INTERNAL PRESSURE/MODUL
1 US OF ELASTICITY,//5X,15HF1CRITICAL = ,D15.5,10X,15HF2CRITICAL
2 = ,D15.5//)
 GO TO 665
662 CONTINUE
 IF(KPAR1.NE.4)GO TO 665
666 WRITE(6,666)N,MT,KHAT,ZMOM,ZRATIO,F1N,F2N,ALIT(N),I
 FORMAT(11X,I4,2X,I4,7X,I5,2X,D12.5,4D15.5,10X,I5//)
 IF(MT.NE.MC)GO TO 665
667 WRITE(6,667)F1CRIT,F2CRIT
 FORMAT(//15X,46HPARAMETER INVESTIGATED = UNIT THERMAL STRAIN,/15X
1,15HF1CRITICAL = ,D15.5,10X,15HF2CRITICAL = D15.5//)
 GO TO 665
665 CONTINUE

TEST TO SEE IF ALL COMBINATIONS HAVE BEEN CONSIDERED FOR THE
CURRENT VALUE OF THE PARAMETER. IF THEY HAVE, GO TO THE NEXT PARA
METER VALUE UNLESS ALL VALUES OF THE RANGE HAVE BEEN INVESTIGATED.
IF THEY HAVE NOT, CHANGE TO NEXT COMBINATION AND MAKE ALL CALCULAT
IONS AS BEFORE.

IF(MT.EQ.MC)GO TO 668
MT=MT+1
GO TO 709
668 CONTINUE
 IF(J.NE.1)GO TO 671
 IF(MD.EQ.1.CR.MD.EQ.3)GO TO 669
 GO TO 671
669 WRITE(6,670)EXMOM(N)
670 FORMAT(40X,18HEXACT MOMENT = D15.5//)
671 CONTINUE

TEST TO SEE IF ALL VALUES OF THE PARAMETER HAVE BEEN INVESTIGATED.
IF THEY HAVE NOT, INCREMENT THE PARAMETER TO THE NEXT VALUE IN
THE RANGE. IF THEY HAVE, GO TO THE DRAW SUBROUTINE IF THE

```

CC  
CC  
CC  
CC  
CC  
CC

CC  
CC  
CC  
CC







# E-2 Subroutine ALPHA

```

SUBROUTINE ALPHA(X,C,EPI,I1,I2,I3,I4,I5,I6,I7,I8,I9,I10)
IMPLICIT REAL*8(A-H,O-Z)
IF(X-0.0)4,6,5
IF(X+EPI)7,6,6
IF(X-EPI)6,6,8
I1=0.5+(5.0/24.0)*X+((61.0/720.0)*(X**2))+((277.0/8064.0)*(X**3))
I2=1.0/3.0+(2.0/15.0)*X+(17.0/315.0)*(X**2)+(62.0/2835.0)*(X**3)
I3=1.0+X*I2
I4=1.0+X*I1
I5=I1+I2+X*I1*I2-5.0/12.0-(61.0/360.0)*X-(277.0/4032.0)*(X**2)-(5052.0/6330.0)*X**3
I6=2.0*I2+X*(I2**2)-.4-(17.0/105.0)*X-(62.0/945.0)*(X**2)-(17900.0/6330.0)*X**3
I66=0.0/675675.0*(X**3)
GO TO 9
7 TX=DABS(X)
Y=-DSQRT(TX)
I3=DTANH(Y)/Y
I4=1.0/DCOSH(Y)
I1=(I4-1.0)/X
I2=(I3-1.0)/X
I5=(I3*I4-2.0*C*I1)/X
I6=(I3**2-3.0*I2)*X
GO TO 9
8 Z=DSQRT(X)
I3=DTAN(Z)/Z
I4=1.0/DCOS(Z)
I1=(I4-1.0)/X
I2=(I3-1.0)/X
I5=(I3*I4-2.0*C*I1)/X
I6=(I3**2-3.0*I2)/X
I7=I4**2+I3-2.0
I8=I3**2-I2
I9=I5+C*I3*I4
I10=I6+2.0*C*I8+(C**2)*I7
END

```

# E-3 Subroutine FACTOR

```

SUBROUTINE FACTOR (JT,R1,R2,R3,R4,LLAT)
 IF(JT-1) 11, 11, 20
11 R1=1.0
 R2=1.0
 R3=1.0
 R4=1.0
 LLAT=1111
 GO TO 999
20 IF(JT-2) 21, 21, 30
21 R1=1.0
 R2=0.0
 R3=0.0
 R4=0.0
 LLAT=1000
 GO TO 999
30 IF(JT-3) 31, 31, 40
31 R1=0.0
 R2=1.0
 R3=0.0
 R4=0.0
 LLAT=0100
 GO TO 999
40 IF(JT-4) 41, 41, 50
41 R1=0.0
 R2=0.0
 R3=1.0
 R4=0.0
 LLAT=0010
 GO TO 999
50 IF(JT-5) 51, 51, 60
51 R1=0.0
 R2=0.0
 R3=0.0
 R4=1.0
 LLAT=0001
 GO TO 999
60 IF(JT-6) 61, 61, 70
61 R1=0.0
 R2=0.0
 R3=0.0
 R4=0.0
 LLAT=0000
 GO TO 999
70 IF(JT-7) 71, 71, 80
71 R1=1.0
 R2=0.0
 R3=1.0
 R4=0.0

```

```

00006580
00006590
00006600
00006610
00006620
00006630
00006640
00006650
00006660
00006670
00006680
00006690
00006700
00006710
00006720
00006730
00006740
00006750
00006760
00006770
00006780
00006790
00006800
00006810
00006820
00006830
00006840
00006850
00006860
00006870
00006880
00006890
00006900
00006910
00006920
00006930
00006940
00006950
00006960
00006970
00006980
00006990
00007000
00007010
00007020
00007030
00007040
00007050

```

```

LLAT= 1010
GC TO 999
80 IF(JT-8) 81, 81, 90
81 R1=0.0
R2=1.0
R3=1.0
R4=0.0
LLAT= 0110
GC TO 999
90 IF(JT-9) 91, 91, 110
91 R1=0.0
R2=0.0
R3=1.0
R4=1.0
LLAT= 0011
GC TO 999
110 IF(JT-10) 111, 111, 120
111 R1=0.0
R2=1.0
R3=0.0
R4=1.0
LLAT= 0101
GC TO 999
120 IF(JT-11) 121, 121, 130
121 R1=1.0
R2=0.0
R3=0.0
R4=1.0
LLAT= 1001
GC TO 999
130 IF(JT-12) 131, 131, 140
131 R1=0.0
R2=1.0
R3=1.0
R4=1.0
LLAT= 0111
GC TO 999
140 IF(JT-13) 141, 141, 150
141 R1=1.0
R2=0.0
R3=1.0
R4=1.0
LLAT= 1011
GC TO 999
150 IF(JT-14) 151, 151, 160
151 R1=1.0
R2=1.0
R3=0.0

```

```

00007060
00007070
00007080
00007090
00007100
00007110
00007120
00007130
00007140
00007150
00007160
00007170
00007180
00007190
00007200
00007210
00007220
00007230
00007240
00007250
00007260
00007270
00007280
00007290
00007300
00007310
00007320
00007330
00007340
00007350
00007360
00007370
00007380
00007390
00007400
00007410
00007420
00007430
00007440
00007450
00007460
00007470
00007480
00007490
00007500
00007510
00007520
00007530

```



```

R4=1.0 1101
LLAT= 999
GC TC 999
160 IF(JT-15)161,161,170
161 R1=1.0
R2=1.0
R3=1.0
R4=0.0 1110
LLAT= 999
GO TC 999
170 IF(JT-16)171,171,999
171 R1=1.0
R2=1.0
R3=0.0
R4=0.0 1100
LLAT= 1100
999 END

```

```

00007540
00007550
00007560
00007570
00007580
00007590
00007600
00007610
00007620
00007630
00007640
00007650
00007660
00007670
00007680
00007690
00007700

```

E-4 Sample Output

| A        |          | B          | CIN         | DOUT        | DTEMP       | P           | E               | COEFL       | KPAR1 | ITERATIONS |
|----------|----------|------------|-------------|-------------|-------------|-------------|-----------------|-------------|-------|------------|
| N        | MT       | DESIGNATOR | MOMENT      | MOMRATIO    | FIN         | F2N         | PARAMETER VALUE |             |       |            |
| 0.36D 03 | 0.36D 03 | 0.36D 03   | 0.10D 02    | 0.11D 02    | 0.53D 03    | 0.40D 03    | 0.27D 08        | 0.723D-05   | 1     |            |
|          | 21       | 1          | 1111        | 0.107150 07 | 0.100000 01 | 0.658410 05 | 0.463390 04     | 0.200000 00 | 2     |            |
|          | 21       | 2          | 1000        | 0.121660 07 | 0.113540 01 | 0.741700 05 | 0.514380 04     | 0.200000 00 | 2     |            |
|          | 21       | 3          | 100         | 0.120630 07 | 0.112580 01 | 0.715540 05 | 0.509680 04     | 0.200000 00 | 2     |            |
|          | 21       | 4          | 10          | 0.116460 07 | 0.108680 01 | 0.708190 05 | 0.493090 04     | 0.200000 00 | 2     |            |
|          | 21       | 5          | 1           | 0.115670 07 | 0.107950 01 | 0.733080 05 | 0.502370 04     | 0.200000 00 | 2     |            |
|          | 21       | 6          | 0           | 0.123110 07 | 0.114900 01 | 0.749720 05 | 0.520860 04     | 0.200000 00 | 2     |            |
|          | 21       | 7          | 1010        | 0.115220 07 | 0.107530 01 | 0.701410 05 | 0.487560 04     | 0.200000 00 | 2     |            |
|          | 21       | 8          | 110         | 0.114390 07 | 0.106760 01 | 0.677620 05 | 0.483700 04     | 0.200000 00 | 2     |            |
|          | 21       | 9          | 11          | 0.109930 07 | 0.102600 01 | 0.694140 05 | 0.476790 04     | 0.200000 00 | 2     |            |
|          | 21       | 10         | 101         | 0.113710 07 | 0.106120 01 | 0.700620 05 | 0.492540 04     | 0.200000 00 | 2     |            |
|          | 21       | 11         | 1001        | 0.114290 07 | 0.106660 01 | 0.724820 05 | 0.496160 04     | 0.200000 00 | 2     |            |
|          | 21       | 12         | 111         | 0.108300 07 | 0.101070 01 | 0.664950 05 | 0.468520 04     | 0.200000 00 | 10    |            |
|          | 21       | 13         | 1011        | 0.108750 07 | 0.101490 01 | 0.687120 05 | 0.471470 04     | 0.200000 00 | 2     |            |
|          | 21       | 14         | 1101        | 0.112390 07 | 0.104890 01 | 0.693000 05 | 0.486630 04     | 0.200000 00 | 2     |            |
|          | 21       | 15         | 1110        | 0.113200 07 | 0.105650 01 | 0.671310 05 | 0.478390 04     | 0.200000 00 | 2     |            |
| 21       | 16       | 1100       | 0.119250 07 | 0.111290 01 | 0.708160 05 | 0.503520 04 | 0.200000 00     | 5           |       |            |

PARAMETER INVESTIGATED = LEG RATIO A/B

EXACT MOMENT = 0.107150 07

ELAPSED TIME = 16.05

APPENDIX F  
SAMPLE CALCULATION

The following table of values is provided to aid the user in checking the computer calculations. Two combinations of secondary effects are listed. The first is the "Exact Value" combination where all secondary effects are included. The second is the most erroneous combination, or the one where all secondary effects are suppressed in the calculation. The descriptive data is that listed on page 42 of Section IV and also exactly the same as the sample data card in Appendix C, with a leg ratio of 0.2.

| <u>Quantity</u>  | <u>All Effects Allowed</u> | <u>All Effects Suppressed</u> |
|------------------|----------------------------|-------------------------------|
| Assumed F1       | 65372.66553                | 74373.10291                   |
| Assumed F2       | 4623.212071                | 5192.148178                   |
| W                | .0054419338                | 0.0                           |
| $\alpha_1$       | .5011360506                | .5000000000                   |
| $\alpha_2$       | .3340605064                | .3333333135                   |
| $\alpha_3$       | 1.001817935                | 1.0000000000                  |
| $\alpha_4$       | 1.002727150                | 1.0000000000                  |
| $\alpha_5$       | .4185170188                | .4166666865                   |
| $\alpha_6$       | .2678453147                | .2666666508                   |
| $\alpha_7$       | .0072796731                | 0.0                           |
| $\alpha_8$       | .6695786688                | .6666666865                   |
| $\alpha_9$       | .4320902252                | .4166666865                   |
| $\alpha_{10}$    | .2859409726                | .2666666508                   |
| WBAR             | 1.923736751                | 0.0                           |
| $\bar{\alpha}_1$ | 2.324235399                | .5000000000                   |
| $\bar{\alpha}_2$ | 1.496163752                | .3333333135                   |

| Quantity            | All Effects Allowed | All Effects Suppressed |
|---------------------|---------------------|------------------------|
| $\bar{\alpha}_3$    | 3.878225194         | 1.000000000            |
| $\bar{\alpha}_4$    | 5.471216862         | 1.000000000            |
| $\bar{\alpha}_5$    | 8.613517662         | .4166666865            |
| $\bar{\alpha}_6$    | 5.485230450         | .2666666508            |
| $\bar{\alpha}_7$    | 31.81243915         | 0.0                    |
| $\bar{\alpha}_8$    | 13.54446691         | .6666666865            |
| $\bar{\alpha}_9$    | 8.624985666         | .4166666865            |
| $\bar{\alpha}_{10}$ | 5.499880474         | .2666666508            |
| $\beta_1$           | 1.817163214         | .6000000000            |
| $\beta_2$           | 205.4115482         | 150.0000000            |
| $g_{11}$            | -.0001242864        | -.0000882817           |
| $g_{12}$            | .0018232014         | .0013242253            |
| $g_{21}$            | .0000294904         | .0000247189            |
| $g_{22}$            | -.0001211686        | -.8828170315           |
| $\lambda$           | .0132454070         | 0.0                    |
| $\bar{\lambda}$     | .0132186939         | 0.0                    |
| F1 Converged        | 65840.74973         | 74972.17486            |
| F2 Converged        | 4633.905400         | 5208.593999            |
| Moment              | 1071501.071         | 1231122.149            |

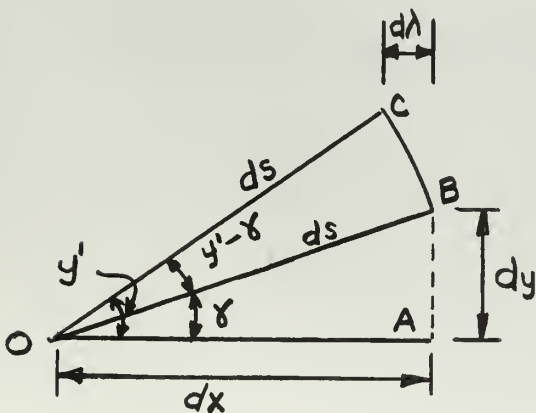
## APPENDIX G

### SHORTENING DUE TO DIFFERENCE BETWEEN ARC AND CHORD

The secondary effect produced because of difference between arc and chord is described by equation (7) of Section II when higher order terms are neglected. The following is a more detailed development of that relationship.

In order to have a physical understanding of the background of the derivation, refer to Fig. G-1. Due to the action of cross shear, as it is idealized in customary theory, the horizontal length,  $dx$ , from 0 to A, is stretched as it is inclined to angle  $\gamma$ , so that it goes from 0 to B. Then the action of bending further inclines it so that it takes the position OC, but there is no change in length during this bending process. (Change in length due to axial force is accounted for separately.) The increment of difference between arc and chord is the distance  $d\lambda$  shown in the figure.

In the derivation, it is first assumed that  $\gamma$  is small so that  $\tan \gamma = \gamma$  approximately.



$$ds^2 = dx^2 + dy^2$$

$$d\lambda = dx - ds \cos(\gamma')$$

$$\frac{dy}{dx} = \gamma \text{ or } dy = \gamma dx$$

$$ds^2 = dx^2 + \gamma^2 dx^2$$

$$ds^2 = dx^2 (1 + \gamma^2)$$

$$\therefore ds = dx \sqrt{1 + \gamma^2}$$

$$d\lambda = dx - dx \sqrt{1 + \gamma^2} \cos(\gamma')$$

$$d\lambda = dx (1 - \sqrt{1 + \gamma^2} \cos(\gamma'))$$



If series representation of  $\cos(y')$  and  $\sqrt{1+y'^2}$  is substituted with higher order terms neglected

$$d\lambda = dx(1 - (1 + \frac{1}{2}y'^2)(1 - \frac{1}{2}(y')^2)) = \frac{dx}{2} ((y')^2 - y'^2)$$

or

$$(7) \quad \lambda = \frac{1}{2} \int_0^L [(y')^2 - y'^2] dx$$

# INITIAL DISTRIBUTION LIST

|                                                                                                                                                    | No. Copies |
|----------------------------------------------------------------------------------------------------------------------------------------------------|------------|
| 1. Defense Documentation Center<br>Cameron Station<br>Alexandria, Virginia 22314                                                                   | 20         |
| 2. Library<br>Naval Postgraduate School<br>Monterey, California 93940                                                                              | 2          |
| 3. Mechanical Engineering Department<br>Naval Postgraduate School<br>Monterey, California 93940                                                    | 1          |
| 4. Professor J. E. Brock<br>Department of Mechanical Engineering<br>Naval Postgraduate School<br>Monterey, California 93940                        | 4          |
| 5. Mr. Stanley Poley<br>IBM<br>New York Scientific Center<br>590 Madison Avenue, North<br>New York, New York 10022                                 | 1          |
| 6. Dr. L. H. Chen<br>Electric Boat Division<br>General Dynamics Corporation<br>Groton, Connecticut 06340                                           | 1          |
| 7. Mr. James H. Griffin<br>Project Engineering Group<br>Engineering Department<br>Los Alamos Scientific Laboratory<br>Los Alamos, New Mexico 87544 | 1          |
| 8. Mr. Leslie Kaldor<br>Navy Marine Engineering Laboratory<br>Annapolis, Maryland 21402                                                            | 1          |
| 9. Lt. John H. T. Miles<br>5802 Sunmist Drive<br>Palos Verdes Peninsula, California 90274                                                          | 4          |



## Security Classification

## DOCUMENT CONTROL DATA - R&amp;D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

|                                                                                                |  |                                                                             |                 |
|------------------------------------------------------------------------------------------------|--|-----------------------------------------------------------------------------|-----------------|
| 1. ORIGINATING ACTIVITY (Corporate author)                                                     |  | 2a. REPORT SECURITY CLASSIFICATION                                          |                 |
| Naval Postgraduate School<br>Monterey, California                                              |  | Unclassified                                                                |                 |
|                                                                                                |  | 2b. GROUP                                                                   |                 |
| 3. REPORT TITLE                                                                                |  |                                                                             |                 |
| Numerical Study of Secondary Effects in Flexibility Analysis of an Ell-Shaped Piping Structure |  |                                                                             |                 |
| 4. DESCRIPTIVE NOTES (Type of report and inclusive dates)                                      |  |                                                                             |                 |
| Master of Science Thesis                                                                       |  |                                                                             |                 |
| 5. AUTHOR(S) (Last name, first name, initial)                                                  |  |                                                                             |                 |
| MILES, John H. T.                                                                              |  |                                                                             |                 |
| 6. REPORT DATE                                                                                 |  | 7a. TOTAL NO. OF PAGES                                                      | 7b. NO. OF REFS |
| September 1967                                                                                 |  | 94                                                                          | 9               |
| 8a. CONTRACT OR GRANT NO.                                                                      |  | 8a. ORIGINATOR'S REPORT NUMBER(S)                                           |                 |
| b. PROJECT NO.                                                                                 |  |                                                                             |                 |
| c.                                                                                             |  | 9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report) |                 |
| d.                                                                                             |  |                                                                             |                 |
| 10. AVAILABILITY/LIMITATION NOTICES                                                            |  |                                                                             |                 |
| [REDACTED]                                                                                     |  |                                                                             |                 |
| [REDACTED]                                                                                     |  |                                                                             |                 |
| [REDACTED]                                                                                     |  |                                                                             |                 |
| 11. SUPPLEMENTARY NOTES                                                                        |  | 12. SPONSORING MILITARY ACTIVITY                                            |                 |
|                                                                                                |  |                                                                             |                 |

## 13. ABSTRACT

This thesis considers the influence of four secondary effects on the analysis of a simple ell-shaped piping structure when it is uniformly heated. The four effects are: (a) axial deformation, (b) shearing deformation, (c) beam-column effect, and (d) difference between arc and chord. They are usually neglected in a piping flexibility analysis.

In a paper of several years ago, J. E. Brock developed a theory for this investigation but did not present any numerical results or conclusions. The present thesis reviews Brock's theory and corrects some errors in coefficients. It then describes a digital computer program for IBM System 360 which performs the corresponding calculations. Finally, it draws the new and quite significant conclusion that conventional piping stress analysis, which neglects these secondary effects, can result in gross errors in evaluating stresses in piping configurations which are likely to be encountered in piping practice.

14

## KEY WORDS

## LINK A

## LINK B

## LINK C

ROLE

WT

ROLE

WT

ROLE

WT

Ell-Shaped Pipe

Secondary Effects

Pipe Expansion

Piping Flexibility

Heating of Pipe

Structural Deformation

Secondary Deformation















thesM58516

Numerical code of

DUDLEY KNOX LIBRARY



3 2768 00416417 8

DUDLEY KNOX LIBRARY